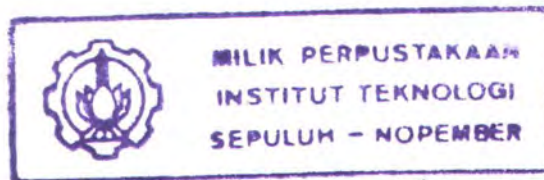


20.811/H/04



IMPLEMENTASI SISTEM PENGENALAN WAJAH
DENGAN PENERAPAN METODE
ADAPTIVE NEURO-FUZZY INFERENCE SYSTEM (ANFIS)

TUGAS AKHIR



RS1f
006.32
Dew
i
2004

PERPUSTAKAAN ITS	
Tgl. Terima	29-7-2004
Terima Dari	H/
Agenda Prp.	220405

Disusun Oleh :

A. A. KUSUMA DEWI

5199 100 027

JURUSAN TEKNIK INFORMATIKA
FAKULTAS TEKNOLOGI INFORMASI
INSTITUT TEKNOLOGI SEPULUH NOPEMBER
SURABAYA
2004

**IMPLEMENTASI SISTEM PENGENALAN WAJAH
DENGAN PENERAPAN METODE
ADAPTIVE NEURO-FUZZY INFERENCE SYSTEM (ANFIS)**

TUGAS AKHIR

Diajukan Untuk Memenuhi Sebagian Persyaratan

Memperoleh Gelar Sarjana Komputer

Pada

Jurusan Teknik Informatika

Fakultas Teknologi Informasi

Institut Teknologi Sepuluh Nopember

Surabaya

Mengetahui / Menyetujui,



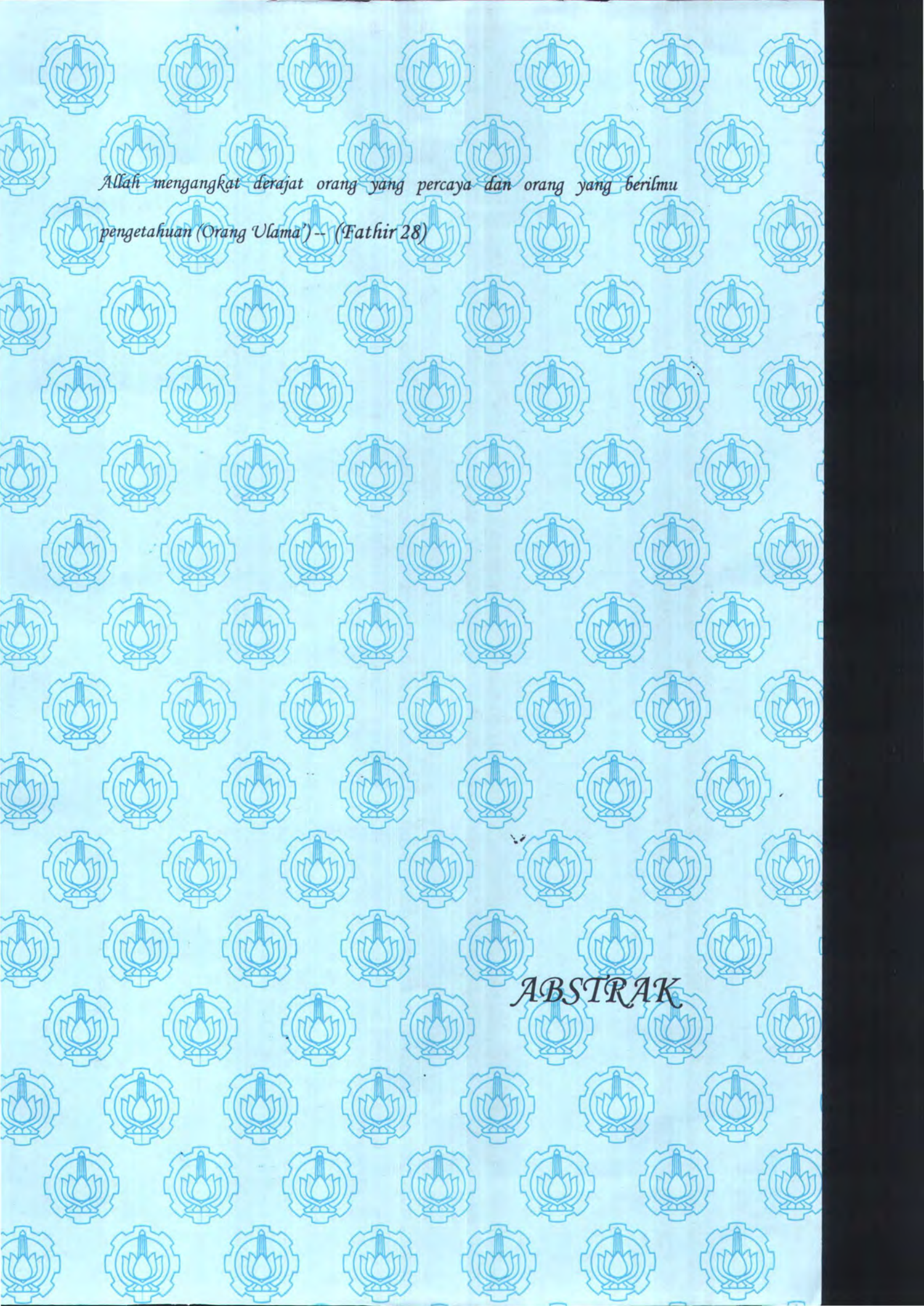
Dosen Pembimbing

Rully Soelaiman, S.Kom, M.Kom

NIP. 132 085 802

SURABAYA

JUNI 2004



*Allah mengangkat derajat orang yang percaya dan orang yang berilmu
pengetahuan (Orang Ulama') -- (Fathir 28)*

ABSTRAK

ABSTRAK

ANFIS adalah metode yang merupakan penggabungan mekanisme Sistem Inferensi Fuzzy yang direpresentasikan dalam arsitektur jaringan syaraf tiruan. Umumnya ANFIS diterapkan sebagai metode pembelajaran pada sistem kendali, dengan karakteristik ANFIS tersebut dapat dilihat adanya peluang untuk menerapkan ANFIS sebagai metode pembelajaran pada klasifikasi.

Diperlukan fitur ekstraktor yang tepat dalam penerapan ANFIS pada sistem pengenalan wajah. Fitur ekstraktor yang dibutuhkan ada dua, yaitu metode untuk mengekstrak fitur global dan fitur lokal. Metode yang tepat untuk mengekstraksi fitur global adalah Analisa Komponen Utama (PCA - Principal Component Analysis) karena mampu mendapatkan basis proyeksi optimal dimana metode ini dimodifikasi dengan transformasi whitening dan untuk mengekstraksi fitur lokal digunakan metode Analisa Paket Wavelet yang akan mendekomposisi citra wajah sehingga menghasilkan informasi lokal wajah. Di dalam ANFIS terdapat dua metode pembelajaran, yaitu backpropagation dengan metode gradient descent untuk arah mundur dan metode Reursive Least Square Estimator untuk arah majunya. Pada metode pembelajaran arah mundur inilah yang nantinya akan merubah parameter premis sedangkan pembelajaran arah maju akan merubah parameter konsekuen.

Sebagai ujicoba utama, perangkat lunak diimplementasikan pada Basis Data baku, yaitu Basis Data Ollivetti, YALE dan BERN. Ujicoba dilakukan terhadap data yang di-whitening dan data yang tidak di-whitening. Untuk data yang tidak di-whitening, hasil akurasi data pembelajaran terhadap Basis Data Ollivetti berkisar antara 50%-53% sedangkan untuk data uji-nya berkisar antara 26%-35% . Ujicoba data pembelajaran terhadap Basis Data YALE berkisar antara 83%- 89% dan untuk data uji berkisar antara 50%- 72. Hasil akurasi data pembelajaran pada percobaan terhadap Basis Data BERN berkisar antara 68% -79% dan akurasi pada data ujinya berkisar antara 44% -57%.

Sedangkan untuk data yang di-whitening, karena melibatkan basis non-orthogonalnya maka hasil akurasinya lebih baik, yaitu hasil akurasi data pembelajaran terhadap Basis Data Ollivetti berkisar antara 98%- 100% dan untuk data uji-nya berkisar antara 71%-80% . Ujicoba data pembelajaran terhadap Basis Data YALE berkisar antara 99%- 100% dan untuk data uji berkisar antara 75.5%-97.7%. Sementara itu, hasil akurasi data pembelajaran pada percobaan terhadap Basis Data BERN berkisar antara 99%- 100% dan akurasi pada data ujinya berkisar antara 80% -100%.

KATA PENGANTAR

Alhamdulillahahirabbil' aalamiin. Segala puji dan syukur semata ditujukan ke hadirat Allah SWT, Tuhan semesta alam, yang telah menciptakan manusia dari segumpal darah dan menjadikan kegelapan dan cahaya. Shalawat serta salam semoga tetap tercurah kepada Rasulullah Muhammad SAW, keluarga, sahabat dan orang-orang yang mengikutinya sampai akhir jaman. Berkat rahmat dan hidayah-Nya pula akhirnya penulis dapat menyelesaikan Tugas Akhir berjudul **"Implementasi Sistem Pengenalan Wajah dengan Penerapan Metode Adaptive-Neuro Fuzzy Inference System "**.

Tugas Akhir ini merupakan salah satu syarat untuk menyelesaikan program strata satu (S-1) pada jurusan Teknik Informatika, Fakultas Teknologi Informasi, Institut Teknologi Sepuluh Nopember Surabaya.

Penulis menyadari bahwa Tugas Akhir ini tidak lepas dari bantuan banyak pihak, maka dalam kesempatan ini, penulis ingin menyampaikan penghormatan dan terima kasih kepada pihak-pihak yang telah memberi bantuan baik itu berupa moril maupun material secara langsung maupun tidak langsung kepada :

1. Bapak Prof. Dr. Ir. Arif Djunaidy, M.Sc, selaku Ketua Dekan Fakultas Teknologi Informasi, ITS.
2. Bapak Yudhi Purwananto, S.Kom, M.Kom, selaku Ketua Jurusan Teknik Informatika, FTIF, ITS.

3. Bapak Rully Soelaiman, S.Kom, M.Kom, selaku Dosen Pembimbing tunggal, yang selalu memberikan bimbingan, petunjuk yang berarti, dan pelajaran “lain” yang tidak diberikan dosen lain, juga atas semua bantuan yang diberikan kepada penulis.
4. Bapak Ir. F.X. Arunanto, M. Sc, selaku koordinator lab- komputing, yang telah mengijinkan penulis mengerjakan Tugas Akhir ini sampai selesai.
5. Bapak dan Ibu dosen Jurusan Teknik Informatika – ITS yang telah dengan sabar memberikan ilmunya selama Penulis menempuh kuliah.
6. Seluruh staf dan karyawan Jurusan Teknik Informatika – ITS, Mas Yudi, Pak Sugeng, Pak Narno, Pak Mu’in, Mbak Eva, Pak Karmono, Bu Tuti dan karyawan lain.
7. Ibu yang sudah berusaha membesarkan anak-anaknya dengan penuh kesabaran dan tanggung jawab-‘hanya Allah yang dapat membalas semua kebaikan dan kesabaran ibu’, bapak yang selalu mendoakan penulis dan juga atas aliran dananya ☺. *(Ya Allah..ampunilah mereka, kasih sayangilah mereka, sebagaimana mereka mengasihi aku waktu kecil)*
8. Mas Bowo yang nun jauh disana tapi dekat dihati, Mas Uton yang dengan penuh kesabaran dan “pengganti bapak”pelindung buat adik2-nya- ‘Dewi dah jadi S. Kom mas...berkat mas Uton’, Mas Yono yang slalu menghibur, mau mendengarkan keluh kesah penulis, yang selalu memperhatikan penulis- ‘disetiap

cobaan selalu membawa hikmah yang indah ya mas....', Mas Antok yang dengan diamnya mampu memberikan kasih sayang buat adek2, Mas Yongki yang selalu sabar dan setia mendampingi penulis juga atas curahan kasih sayang yang telah diberikan.

9. Adek2 penulis yang manis, Minggar yang sabar mendengar dan menghibur penulis, Iis yang selalu memberi semangat penulis dan atas sms2 lucunya juga, Hevy atas doa, dorongan semangat dan perhatiannya kepada penulis.
10. Seluruh keluarga kesar di Solo, Simbah (mbah putri), Budhe Marmi, Budhe Saminem, Budhe Lasinem, M'Jarwo, M'Sri, M' Bambang, M'Gun, M'Usman, M'Suli, M'Mimin, M'Esti, M'Pur, M'Kis, M'Wati, M' Tri, M'No, M'Sri, M'Pur, M'Encik, dan sodara2 laen
11. C-11 plus plus, Ilma yang selalu ada saat penulis butuhkan dan dengan penuh kesabaran membantu penulis-'Cuma kamu yang ngerti Taku ma, setelah mendengar presentasiku 5 kali + ngantuk2, bruntung bgt yg "dapat" kamu', Ridha yang selalu ceria, yang dapat membangkitkan semangat penulis, bener2 temen senasib seperjuangan, Nanoom yang selalu duduk didekat penulis yang hadir dengan taujihnya dan merubah pandangan penulis tentang arti hidup,-'adem rasanya dekat kamu num'- orion nebula jadi saksi kita, M'Nuning yang memberi inspirasi penulis untuk penuh semangat menjalani hidup- j2, Eva yang selalu sabar dan pasrah-'kerjain TA-nya va... jgn jd anak SMA trs.:P', Dewi atas nasehat dan cerita pengalamannya, Intan yang tak bosan-bosannya memberikan

dukungan dan bantuan kepada penulis dari proposal sampai sidang TA-‘kmu yg ngenalin aku ma Papi tan, bruntung banget bisa menimba ilmu dari beliau ‘, Audrey atas dorongan semangatnya.

12. Teman-teman TA-ers, ririd yang sabar-‘Chayo!! Chayo!!’. Ana “pejuang ulet”-yang selalu menyemangati penulis, Adeta “si optimis” -atas kue2nya dan kata2 optimis yang dapat membuat penulis bangkit, Budi atas dorongan semangatnya-‘kerjain TAny bud, jgn pemilu trs yang diurusin’, CD-Key atas smua bantuannya, wa2n atas bantuannya menjelang sidang-‘thx dah ngambilin semua keperluan sidang’, hendra-‘ thx buat mouse-nya’, Agung, Heru, Alwi.
13. Teman-teman seperjuangan, m’Alfi atas doa-doa dan taujihnya, Anik atas bantuan juga semangat yang selalu diberikan saat penulis butuhkan, Dian, Nina atas perhatian, taujih dan kesabarannya dalam menyemangati penulis, Mimik atas perhatian dan doanya, Cici’ atas bantuan moril dan perhatiannya selama penulis kuliah sampai sidang TA- ‘dulu dimana ada kmu disitu ada aku ci’ ☺..seiring bjalannya wkt...kita skr jrg bareng lagi.. anyway uhibbuki fillah ukhti’, m’ Rien atas doanya, Anny atas transfer semangatnya, m”Hendrat, m’Ida, Sofi00, Lintang00, Ruli00, Yunita00, Putri00, Rahmi00, Fetty00, feby02, ikhwana02, dll
14. J_ers99, yupee-ndul atas cerita2nya- ‘dibalik semua itu pasti ada hikmahnya..jd lbh fresh abis denger critamu ndul’ , sunarsih atas doanya, triyas atas dorongan semangatnya, dwi atas perhatian dan persahabatannya, Luky atas persahabatan

dan kasih sayangnya, andik-bohood atas semangat dan keywordnya- 'keywordmu siip ndik, waveletku akhirnya jadi' dan temen2 jember yg laen.

15. Teman-teman 99-ers, Medi, Farida, Nur- 'Bukuku no revisi nur, berkat bantuan nge-printmu ☺', Rizky Ika, I'an, Meme, double_R, Anton, Affie, Firman, Eri, Rully, Dony, Erfan, Indie atas pinjaman printernya, Ade Reza, Cherry, Kamal, Rifqi, Wahib, Chendra, Arie, AW, Rontog, Liga
16. Komputing-ers, mas "Admin" Gershom yang selalu membantu dengan penuh kesabaran, mas Agus yang selalu nggodain dan membantu penulis-'Ayo mas, jangan nggodain terus, kapan nih sidangnya...orion punya mas Agus deh', mbak Luluk, mbak Faida-'Sabar itu susah mbak.. tapi buahnya manis', mas Kilay, mbak Dwi, mbak Tutik, Eddy atas perhatian dan traktirannya ☺, Fandy, Connie, Kadek.
17. Serta rekan-rekan lain yang tidak dapat disebutkan satu-persatu yang telah membantu penulis mulai dari awal kuliah hingga tersusunnya tugas akhir ini.

Penulis berharap semoga apa yang telah penulis uraikan dapat bermanfaat dan dapat dikembangkan lebih lanjut untuk perkembangan ilmu pengetahuan dan teknologi dimasa mendatang.

Surabaya, Juni 2004

Anak Agung Kusuma Dewi

DAFTAR ISI

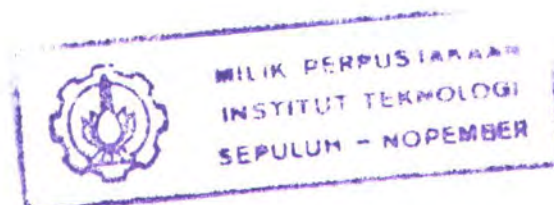
ABSTRAK.....	i
KATA PENGANTAR.....	vii
DAFTAR ISI.....	vii
DAFTAR GAMBAR.....	x
DAFTAR TABEL.....	xii
BAB I	1
PENDAHULUAN.....	1
1.1. LATAR BELAKANG	1
1.2. TUJUAN	2
1.3. PERMASALAHAN	3
1.4. METODOLOGI.....	3
1.5. SISTEMATIKA PENULISAN	4
BAB II.....	7
DASAR TEORI	7
2.1 ANALISA KOMPONEN UTAMA (PCA).....	7
2.1.1 Konstruksi Vektor Sebuah Gambar	8
2.1.2 Ruang Lingkup Gambar.....	8
2.1.3 Transformasi Matrik	9
2.1.4 Matriks Kovariansi	9
2.1.5 Nilai Eigen dan Vektor Eigen	10
2.1.6 Reduksi Dimensi.....	11
2.1.7 Transformasi Whitening (<i>Whitening Transformation</i>)	12
2.2 ANALISA PAKET WAVELET	15

2.2.1	Transformasi Wavelet.....	16
2.2.2	Analisa Multiresolusi.....	17
2.2.3	Konvolusi.....	18
2.2.4	Dekomposisi.....	18
2.2.5	Rekonstruksi.....	23
2.3	SISTEM INFERENSI FUZZY.....	26
2.3.2	Jaringan Adaptif.....	31
BAB III.....		34
PENERAPAN ANFIS PADA PENGENALAN WAJAH.....		34
3.1	STRUKTUR ANFIS.....	34
3.2	PROSES PEMBELAJARAN PADA ANFIS.....	37
3.2.1	Proses Pembelajaran Backpropagation-error untuk Parameter Premis.....	38
3.2.2	Proses Pembelajaran dengan RLSE untuk Parameter Konsekuen.....	43
BAB IV.....		45
IMPLEMENTASI ANFIS.....		45
4.1	LINGKUNGAN PEMBANGUNAN APLIKASI ANFIS.....	45
4.2	MASUKAN SISTEM PENGENALAN WAJAH.....	45
4.3	PROSES.....	47
4.3.1	Proses Pembacaan Path file lokasi Gambar.....	48
4.3.2	Proses Pembacaan Gambar dan diubah ke bentuk Matrik.....	48
4.3.3	Proses Utama.....	48
4.4	KELUARAN SISTEM PENGENALAN WAJAH.....	58
4.5	ANTARMUKA.....	59
4.5.1	Form Masukan.....	60
4.5.2	Form Keluaran.....	61
BAB V.....		64
UJI COBA DAN EVALUASI.....		64
5.1	LINGKUNGAN UJI COBA.....	64

5.2	PELAKSANAAN UJI COBA DAN EVALUASI.....	64
5.2.1	Uji Kebenaran	65
5.2.2	Uji Kinerja	66
5.2.1	Hasil Ujicoba Data Whitening.....	66
5.2.2	Hasil Ujicoba Data tanpa whitening	71
5.2.3	Hasil Ujicoba Data Whitening.....	76
5.2.4	Hasil Ujicoba Data tanpa Whitening	79
5.2.2	Hasil Ujicoba Data Whitening.....	82
5.2.3	Hasil Ujicoba Data tanpa whitening	86
BAB VI	90
KESIMPULAN DAN SARAN	90
6.1	KESIMPULAN	90
6.2	SARAN	91
DAFTAR PUSTAKA	92
LAMPIRAN A- BASIS DATA WAJAH UNIVERSITY OF BERN	A1
LAMPIRAN B- BASIS DATA WAJAH YALE	B1
LAMPIRAN C- BASIS DATA WAJAH OLLIVETTY	C1

DAFTAR GAMBAR

Gambar 2. 1.	Plot data hasil dari PCA	12
Gambar 2. 2.	Plot data hasil PCA dan transformasi <i>whitening</i>	14
Gambar 2. 3.	Skema Dekomposisi.....	19
Gambar 2. 4.	Bagan Paket Wavelet Complete.....	23
Gambar 2. 5.	Skema Rekonstruksi.....	24
Gambar 2. 6.	Skema Defuzzifikasi untuk mendapatkan nilai keluaran	30
Gambar 2. 7.	Jaringan adaptif Feedforward	33
Gambar 2. 8.	Jaringan adaptif Recurent.....	33
Gambar 3.1.	Struktur ANFIS	35
Gambar 3.2	Model jaringan syaraf feedforward	38
Gambar 3.3	Diagram hubungan sebab akibat	39
Gambar 4.1	Diagram Proses Pengklasifikasian dalam ANFIS	47
Gambar 4.2.	Form Masukan Data	59
Gambar 4.3.	Bagian pengambilan data pada Form Masukan Data.....	60
Gambar 4.4.	Bagian masukan data untuk FIS pada Form Masukan Data	61
Gambar 4.5.	Form Menampilkan Keluaran	61
Gambar 4.6.	Grafik Kesalahan Data- RMSE.....	62
Gambar 4.7.	Grafik Keluaran Sistem.....	62
Gambar 4.8.	Grafik Derajat Keanggotaan	63
Gambar 5.1.	Grafik RMSE untuk Data Ollivetti Leave-10%-out.....	70
Gambar 5.2.	Grafik RMSE untuk Data Ollivetti Leave-25%-out.....	71
Gambar 5.3.	Grafik RMSE untuk Data Ollivetti Leave-50%-out.....	71
Gambar 5.4.	Grafik Akurasi Data Pembelajaran OLLIVETTI.....	75
Gambar 5.5.	Grafik Akurasi Data Uji OLLIVETTI.....	75
Gambar 5.6.	Grafik RMSE untuk Data YALE Leave-10%-out	78



Gambar 5.7.	Grafik RMSE untuk Data YALE Leave-25%-out	78
Gambar 5.8.	Grafik RMSE untuk Data YALE Leave-50%-out	79
Gambar 5.9.	Grafik Akurasi Data Pembelajaran YALE	81
Gambar 5.10.	Grafik Akurasi Data Uji YALE	81
Gambar 5.11.	Grafik RMSE untuk Data BERN Leave-10%-out	85
Gambar 5.12.	Grafik RMSE untuk Data BERN Leave-25%-out	85
Gambar 5.13.	Grafik RMSE untuk Data BERN Leave-50%-out	86
Gambar 5.14.	Grafik tingkat Akurasi Data Pembelajaran BERN	89
Gambar 5.15.	Grafik tingkat Akurasi Data Uji BERN	89

DAFTAR TABEL

Tabel 3.1 Proses pembelajaran hibrid ANFIS.....	37
Tabel 4.1. Lingkungan Pembangunan Sistem Pengenalan Wajah.....	45
Tabel 4.2. Keluaran yang dihasilkan oleh Proses Klasifikasi ANFIS.....	46
Tabel 4.3. Keluaran yang dihasilkan oleh Proses Klasifikasi ANFIS.....	58
Tabel 5.1. Lingkungan Pengujian Sistem Pengenalan Wajah.....	64
Tabel 5.2 Hasil pembelajaran Basis Data BERN dengan Metode ANFIS	65
Tabel 5.3 Hasil pengujian Basis Data BERN dengan Metode ANFIS	66
Tabel 5.4 Hasil pembelajaran Basis Data OLLIVETTI dengan Metode ANFIS.....	67
Tabel 5.5 Hasil pengujian Basis Data OLLIVETTI dengan Metode ANFIS	68
Tabel 5.6 Hasil pembelajaran Basis Data OLLIVETTI (tanpa whitening) dengan Metode ANFIS	72
Tabel 5.7 Hasil pengujian Basis Data OLLIVETTI (tanpa whitening) dengan Metode ANFIS	73
Tabel 5.8 Hasil pembelajaran Basis Data YALE dengan Metode ANFIS.....	76
Tabel 5.9 Hasil pengujian Basis Data YALE dengan Metode ANFIS	77
Tabel 5.10 Hasil pembelajaran Basis Data YALE (tanpa whitening) dengan Metode ANFIS	79
Tabel 5.11 Hasil pengujian Basis Data YALE (tanpa whitening) dengan Metode ANFIS	80
Tabel 5.12 Hasil pembelajaran Basis Data BERN dengan Metode ANFIS.....	82
Tabel 5.13 Hasil pengujian Menggunakan Data BERN dengan Metode ANFIS	83
Tabel 5.14 Hasil pembelajaran Basis Data BERN (tanpa whitening) dengan Metode ANFIS	86
Tabel 5.15 Hasil pengujian Menggunakan Data BERN (tanpa whitening) dengan Metode ANFIS	87

Ibnu Mas'ud r.a. berkata: Rasulullah SAW. Bersabda: Tidak boleh menginginkan kepunyaan orang lain melainkan dua macam. Orang yang diberi oleh Allah kekayaan maka dipergunakan untuk membela haq kebenaran dan orang yang diberi oleh Allah ilmu pengetahuan, hikmat maka diajarkan kepada semua orang – Bukhari, Muslim

BAB I

PENDAHULUAN

BAB I

PENDAHULUAN

Pada bab ini akan dijelaskan tentang latar belakang yang mendasari tugas akhir ini, permasalahan, batasan masalah, tujuan dari pembuatan perangkat lunak, metodologi pembuatan dan sistematika penulisan buku tugas akhir.

1.1. Latar Belakang

Pengenalan wajah merupakan salah satu pendekatan pengenalan pola untuk keperluan identifikasi personal. Pengenalan citra wajah berhubungan dengan obyek yang tidak pernah sama, karena adanya bagian-bagian yang dapat berubah. Perubahan ini dapat disebabkan oleh ekspresi wajah, intensitas cahaya dan sudut pengambilan gambar, atau perubahan asesoris pada wajah. Dalam kaitan ini, obyek yang sama dengan beberapa perbedaan tersebut harus dikenali sebagai satu obyek yang sama.

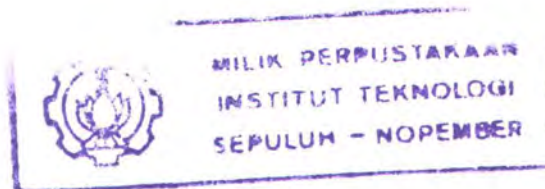
ANFIS adalah metode yang merupakan penggabungan mekanisme Sistem Inferensi Fuzzy yang digambarkan dalam arsitektur jaringan syaraf. Umumnya ANFIS diterapkan sebagai metode pembelajaran pada sistem kendali, dengan karakteristik ANFIS tersebut dapat dilihat adanya peluang untuk menerapkan ANFIS sebagai metode pembelajaran pada klasifikasi khususnya pengenalan wajah.

Ada dua bagian yang penting dalam sistem pengenalan pola, yaitu fitur ekstraktor dan mesin pembelajaran. Untuk menerapkan ANFIS pada sistem pengenalan

wajah, diperlukan fitur ekstraktor yang tepat karena jika ANFIS diterapkan sebagai fitur ekstraktor sekaligus sebagai mesin pembelajarannya maka spesifikasi obyek tidak bisa ditampilkan dan biaya komputasinya akan sangat mahal. Fitur ekstraktor yang dibutuhkan ada dua, yaitu metode untuk mengekstrak fitur global dan fitur lokal [1]. Metode yang tepat untuk mengekstraksi fitur global adalah Analisa Komponen Utama (PCA- *Principal Component Analysis*) karena mampu mendapatkan basis proyeksi optimal dimana metode ini dimodifikasi dengan transformasi whitening dan untuk mengekstraksi fitur lokal digunakan metode Analisa Paket Wavelet yang akan mendekomposisi citra wajah sehingga menghasilkan informasi lokal wajah. Di dalam ANFIS, proses pembelajaran yang digunakan ada dua metode, yaitu penjararan balik (*backpropagation*) dengan metode turunan *gradient* (*gradient descent*) untuk arah mundur dan metode Galat Kuadrat Terkecil (RLSE- *Requursive Least Square Estimator*) untuk arah majunya. Pada metode pembelajaran arah mundur inilah yang nantinya akan merubah parameter premis sedangkan pembelajaran arah maju akan merubah parameter konsekuen.

1.2. Tujuan

Tujuan dari tugas akhir ini adalah merancang dan membuat suatu aplikasi yang menerapkan metode *Adaptive Neuro-Fuzzy Inference System* (ANFIS) sebagai mesin pembelajaran pada sistem pengenalan wajah yang dapat mengkombinasikan hasil ekstraksi fitur global dan fitur lokal.



1.3. Permasalahan

Permasalahan yang diangkat dalam tugas akhir ini adalah:

1. Bagaimana menerapkan metode Analisa Komponen Utama (PCA) sebagai pengekstraksi fitur global dan wavelet sebagai pengekstraksi fitur lokal dalam sistem pengenalan wajah.
2. Bagaimana menerapkan metode *Adaptive Neuro-Fuzzy Inference System* (ANFIS) sebagai mesin pembelajaran pada sistem pengenalan wajah yang dapat mengkombinasikan fitur global dan fitur lokal hasil dari ekstraksi dari Analisa Komponen Utama dan Wavelet.

1.4. Metodologi

Pembuatan tugas akhir ini dilakukan dengan mengikuti metodologi sebagai berikut:

1. Studi literatur

Mencari dan mempelajari referensi-referensi yang berkaitan dengan Analisa Komponen Utama (*Principal Component Analysis*), Analisa paket *Wavelet 2-D* dan *Adaptive Neuro-Fuzzy Inference System* (ANFIS). Pembelajaran ini didapat baik dari buku-buku literatur maupun beberapa artikel di internet. Selain itu dipelajari juga bahasa pemrograman yang akan digunakan, yaitu Matlab 6.1

2. Pengembangan Algoritma

Pada tahap ini dirancang penerapan algoritma yang akan digunakan pada perancangan perangkat lunak yaitu Analisa Komponen Utama (PCA), Analisa Paket Wavelet dan *Adaptive Neuro-Fuzzy Inference System* (ANFIS)

3. Perancangan Perangkat Lunak

Pada tahap ini akan dilakukan perancangan terhadap perangkat lunak yang meliputi data yang akan digunakan, proses-proses yang akan dilaksanakan dan penentuan rancangan antar muka berdasarkan studi pustaka yang telah dilakukan.

4. Pembuatan Perangkat Lunak

Pada tahap ini, perangkat lunak dibuat sesuai dengan desain yang telah dirancang sebelumnya. Data yang digunakan, proses masukan, algoritma dan keluaran didasarkan pada rancangan yang telah dibuat pada tahap sebelumnya.

5. Uji Coba dan Evaluasi Perangkat Lunak

Pada tahap ini, program yang telah dibuat diuji kebenarannya menggunakan aplikasi klasifikasi.

6. Penyusunan Buku tugas akhir

Pada tahap ini dilakukan seiring dengan pengerjaan tugas akhir ini. Penulisan dokumentasi meliputi dasar teori dari sistem, desain sistem algoritma yang digunakan, kinerja sistem yang dibangun serta dokumentasi tahap perancangan dan pembuatan.

1.5. Sistematika Penulisan

Dalam penulisannya, laporan tugas akhir ini dikelompokkan menjadi enam bab, yang membahas tentang bagian-bagian berikut:

BAB I Pendahuluan

Bab ini menjelaskan beberapa hal pokok dari tugas akhir yang telah dilaksanakan, antara lain: latar belakang yang mendasari tugas akhir ini, permasalahan, batasan

masalah, tujuan dari pembuatan perangkat lunak, metodologi pembuatan dan sistematika penulisan buku tugas akhir.

BAB II Dasar Teori

Pada bab ini akan dijelaskan tentang teori-teori yang mendasari tugas akhir ini. Teori itu meliputi metode Analisa Komponen Utama (*Principal Component Analysis*) yang merupakan pengekstraksi fitur global dalam tugas akhir ini, Analisa Paket Wavelet 2 Dimensi yang akan mendekomposisi citra wajah dan bertindak sebagai pengekstraksi fitur lokal. Kemudian dilanjutkan dengan pembahasan tentang Sistem Inferensi Fuzzy.

BAB III Adaptive Neuro-Fuzzy Inference System

Bab ini menjelaskan mengenai penerapan metode *Adaptive Neuro-Fuzzy Inference System* (ANFIS) sebagai mesin pembelajaran pada sistem pengenalan wajah dengan mengkombinasikan fitur global dan fitur lokal hasil dari ekstraksi Komponen Analisa Utama (PCA) dan Analisa Paket Wavelet. Selain itu juga, pada bab ini akan dijelaskan tentang pembelajaran *hybrid* pada ANFIS yang terdiri atas dua bagian yaitu arah maju (*forward pass*) yang menggunakan metode *Recursive Least Square Estimator* (RLSE) dan arah mundur (*backward pass*) yang menggunakan metode *Gradient Descent*. Pembelajaran *hybrid* ini digunakan untuk memperbaharui parameter-parameter premis dan konsekuen.

BAB IV Perancangan dan Pembuatan Perangkat Lunak

Menjelaskan tahap-tahap dalam perancangan dan pembuatan perangkat lunak. Perancangan terdiri dari perancangan data masukan, proses dan antarmuka.

BAB V Uji Coba dan Evaluasi

Pada bab ini dijelaskan mengenai hasil uji coba perangkat lunak untuk proses pembelajaran dan ujicoba ANFIS

BAB VI Penutup

Bab ini adalah bagian terakhir dari buku tugas akhir ini yang berisi kesimpulan dari pembuatan tugas akhir dan saran pengembangan perangkat lunak lebih lanjut.

*Abu Hurairah r.a berkata: Rasulullah SAW. Bersabda: Siapa yang berjalan di
suatu jalan untuk menuntut ilmu pengetahuan, Allah akan memudahkan
baginya jalan ke surga – Muslim*

BAB II

DASAR TEORI

BAB II

DASAR TEORI

Pada bab ini akan dijelaskan tentang teori-teori yang mendasari tugas akhir ini. Teori itu meliputi metode Analisa Komponen Utama (*Principal Component Analysis*) yang akan mereduksi fitur dan merupakan pengekstraksi fitur global, Analisa Paket Wavelet 2 Dimensi yang akan mendekomposisi gambar wajah dan merupakan pengekstraksi fitur lokal. Kemudian dilanjutkan dengan pembahasan tentang Sistem Inferensi Fuzzy.

2.1 ANALISA KOMPONEN UTAMA (PCA)

Bagian penting dalam pengenalan pola atau klasifikasi data adalah fitur ekstraktor yaitu memilih fitur yang mempunyai kemampuan lebih untuk membedakan (fitur ciri) dan mesin pembelajarannya. Pola masukan yang sama seharusnya masuk ke dalam kelas yang sama, sehingga pemilihan fitur ciri sangat penting. Untuk beberapa kumpulan data, gabungan 2 fitur akan mempunyai kemampuan membedakan yang lebih daripada 1 fitur, oleh karena itu kumpulan data kadang-kadang harus diubah ke bentuk yang lebih dapat dikenali atau dipelajari. Komponen Analisa Utama (PCA) adalah metode klasik untuk mendapatkan representasikan data yang terpercaya, berdasarkan pada dekomposisi dari matrik kovarian data masukannya. PCA juga dikenal dengan *Kurhunen Loeve Transformation* dalam teori komunikasi dan proses gambar (*image processing*). Gambar merupakan kumpulan dari piksel-piksel, dimana setiap piksel merupakan

representasi dari angka. Kumpulan dari angka tersebut akan membentuk sebuah matriks.

2.1.1 Konstruksi Vektor Sebuah Gambar

Wajah dapat dipandang sebagai sebuah vektor. Konstruksi vektor dari sebuah gambar dibentuk oleh penggabungan sederhana yaitu dengan meletakkan sebuah baris dari gambar saling bersebelahan dengan baris-baris yang lain. Sehingga sebuah wajah yang merupakan sebuah gambar, dapat dipandang sebagai sebuah vektor. Apabila lebar dan tinggi sebuah gambar adalah w dan h piksel, maka banyaknya komponen dari vektor ini adalah $w \cdot h$. Setiap piksel dikodekan oleh satu komponen vektor.

2.1.2 Ruang Lingkup Gambar

Vektor wajah yang dideskripsikan sebelumnya merupakan bagian dari sebuah ruang. Ruang ini adalah sebuah ruang lingkup gambar (*image space*) yaitu ruang dari semua gambar keseluruhan yang merupakan dimensi $w \cdot h$ piksel. Semua wajah mirip satu sama lain. Mereka sama-sama mempunyai mata, hidung, mulut dan lain-lain yang terletak pada tempat yang sama. Akibatnya semua vektor wajah terletak pada tempat-tempat yang amat berdekatan dalam ruang lingkup gambar. Oleh karena itu, sebuah ruang lingkup gambar yang penuh bukanlah sebuah ruang yang bagus untuk mendeskripsikan sebuah wajah. Vektor basis dari ruang lingkup wajah disebut sebagai komponen utama (*principal component*). Dimensi dari ruang lingkup gambar adalah $w \cdot h$. Tentu saja semua piksel dari sebuah wajah tidak relevan dan setiap piksel bergantung pada tetangganya. Sehingga dimensi

dari ruang lingkup wajah adalah kurang dari dimensi ruang lingkup gambarnya. Dimensi dari ruang lingkup wajah tidak dapat ditentukan tetapi hal ini sudah dapat dipastikan bahwa dimensi dari ruang lingkup wajah akan jauh lebih kecil daripada ruang lingkup gambarnya.

2.1.3 Transformasi Matrik

Diberikan sebuah wajah pelatihan yang terdiri dari K gambar. Ukuran gambar diasumsikan terdiri dari $N \times N$ piksel dimana setiap piksel dikodekan dengan menggunakan 8 bit atau 256 tingkat keabuan (*gray level*). Setiap gambar dapat dipandang sebagai sebuah vektor kolom dengan ukuran $(N \times N) \times 1$.

Misalkan setiap gambar diberi notasi $\Gamma_1, \Gamma_2, \Gamma_3, \dots, \Gamma_k$ yang merupakan vektor kolom $N^2 \times 1$. Bila sebuah gambar dalam wajah pelatihan dinotasikan dalam bentuk matriks, maka akan diperoleh matriks yang berordo $N^2 \times K$.

$\Gamma_n = (\Gamma_1, \Gamma_2, \Gamma_3, \dots, \Gamma_k)^T$, dimana T adalah matrik transpose, sedangkan isi dari tiap vektor kolom adalah

$$\Gamma_n = \begin{bmatrix} \gamma_{1,1} \gamma_{1,2} \dots \gamma_{1,k} \\ \gamma_{2,1} \gamma_{2,2} \dots \gamma_{2,k} \\ \vdots \\ \gamma_{N^2,1} \dots \gamma_{N^2,k} \end{bmatrix} \quad (2.7)$$

2.1.4 Matriks Kovariansi

Matriks kovariansi adalah sebuah matriks simetris berordo $N^2 \times N^2$ dan merupakan sebuah pengukuran yang tidak hanya pada sebuah variansi, tetapi kovariansi dari vektor-vektor kolom.

- Wajah rata-rata, m_x , merupakan operasi rata-rata sederhana dari k kolom yang merepresentasikan sebuah wajah.

$$m_x = \frac{1}{K} \sum_{i=1}^K X_i \quad (2.8)$$

- Deviasi merupakan penyimpangan terhadap rata-rata wajah

$$\Phi_n = \Gamma_n - m_x \quad (2.9)$$

- Misalnya $\Phi_n = \Gamma_n - m_x$ dan $A = (\Phi_1 \ \Phi_2 \ \dots \ \Phi_K)$ adalah matriks wajah ternormalisasi, maka matriks kovariansi di atas dapat ditulis menjadi

$$C_x = A A^T \text{ adalah matriks } \textit{outer product} \text{ berordo } N^2 \times N^2, \text{ atau} \quad (2.10)$$

Jika jumlah data pelatihan K lebih kecil dibandingkan dimensi citra N , maka matriks kovarian dapat dihitung dengan:

$$L = A^T A \text{ adalah matriks } \textit{inner product} \text{ berordo } K \times K, \ K \ll N^2 \quad (2.11)$$

Matriks C_x dan L adalah matriks simetris, sebab unsur-unsur yang berordo pada posisi simetris relative terhadap diagonal utama adalah sama, sehingga dapat diilustrasikan bahwa $\Phi_p^T \Phi_q = \Phi_q^T \Phi_p$, untuk $p, q \in [1, K]$. jadi dapat disimpulkan bahwa nilai matriks $A_{n \times n}$ dikatakan simetris apabila $A^T = A$

2.1.5 Nilai Eigen dan Vektor Eigen

Misal A adalah sebuah matriks $n \times n$. sebuah matrik bukan nol, P yang berukuran $n \times 1$ sedemikian rupa sehingga $AP = \lambda P$ dinamakan vektor eigen bagi A , sedangkan skalar λ dinamakan nilai eigen bagi A yang bersesuaian dengan P .

Persamaan definisinya adalah:

$$AP = \lambda P \text{ ekuivalen dengan}$$

$$\lambda P = AP \text{ atau}$$

$$(\lambda I - A)P = 0 \quad (2.12)$$

persamaan 2.12 akan mempunyai solusi bukan nol jika dan hanya jika:

$$\det(\lambda I - A) = 0 \quad (2.13)$$

2.1.6 Reduksi Dimensi

PCA akan membangkitkan suatu himpunan basis orthonormal yang disebut komponen utama (*principal component*) [3]. Setelah dihasilkan matrik kovarian, selanjutnya matriks vektor eigen dan matriks nilai eigen Φ , Λ dapat dihitung dengan rumus:

$$(A A^T) \Phi = \Phi \Lambda \quad (2.14)$$

Jika ukuran data sample K lebih kecil dibandingkan dimensi N , maka metode berikut dapat diaplikasikan guna menghemat biaya komputasi.

$$(A^T A) \Psi = \Psi \Lambda_1$$

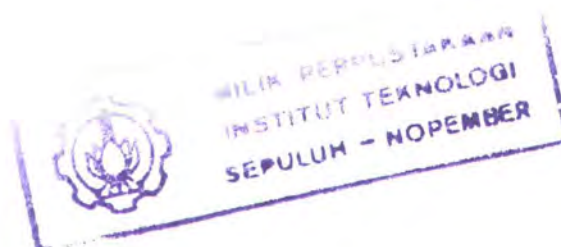
$$\mathfrak{S} = A \Psi \quad (2.15)$$

dengan $\Lambda_1 = \text{diag}\{\lambda_1, \lambda_2, \dots, \lambda_k\}$ dan $\mathfrak{S} = [\Phi_1, \Phi_2, \dots, \Phi_k]$. Jika diasumsikan bahwa nilai eigen telah tersortir secara *decreasing order*, $\lambda_1 \geq \lambda_2 \geq \dots \geq \lambda_k$, maka m vektor eigen terbesar pertama digunakan untuk mendefinisikan matriks P

$$\mathfrak{S} = [\Phi_1, \Phi_2, \dots, \Phi_m] \quad (2.16)$$

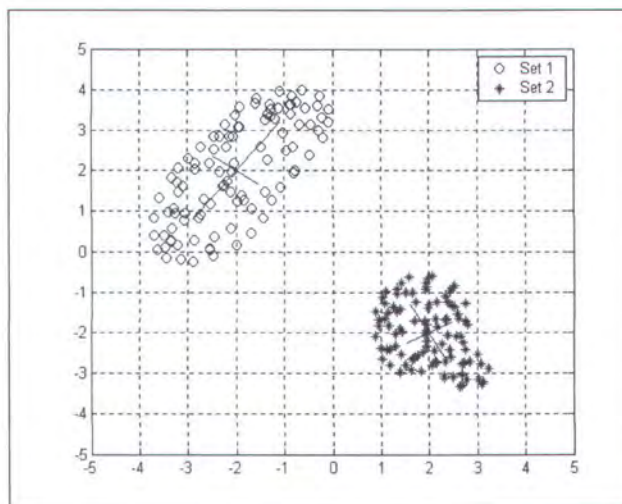
Himpunan ciri baru Z dengan dimensi $m * K$ dapat dihitung dengan rumusan

$$Z = P^T A \quad (2.17)$$



2.1.7 Transformasi Whitening (*Whitening Transformation*)

Dengan menggunakan basis orthogonal sekaligus basis non-orthogonalnya, jangkauan ruang solusi dapat ditingkatkan. PCA membangkitkan suatu himpunan basis orthogonal sehingga jika hasil ini dimodifikasi dengan transformasi *whitening* jangkauan ruang solusi dapat ditingkatkan dan tingkat kesalahan dapat dikurangi. *Whitening* PCA, dalam hal ini mempunyai dua peranan penting, yaitu : mereduksi komponen PCA yang mempunyai bobot relatif kecil, dan meningkatkan jangkauan ruang solusi yang akan melibatkan pula basis non-orthogonal [3]. Berikut adalah contoh penggunaan Transformasi Whitening. Terdapat dua himpunan data dengan aproksimasi mean $\mu_1 = (2,2)$ dan $\mu_2 = (2,-2)$ yang masing-masing mempunyai 100 data sample. Gambar Scatter Plot dari kedua himpunan data tersebut adalah sebagai berikut.



Gambar 2. 1. Plot data hasil dari PCA

Berdasarkan data sample masing-masing himpunan tersebut, selanjutnya dapat dihitung matriks kovariannya.

$$Cov(Set_1) = \begin{bmatrix} 1.04298364190029 & 0.90928280483490 \\ 0.90928280483490 & 1.57004279277764 \end{bmatrix}$$

$$Cov(Set_2) = \begin{bmatrix} 0.39153929738864 & -0.14367586327053 \\ -0.14367586327053 & 0.56636211092900 \end{bmatrix}$$

Selanjutnya, dapat dihitung nilai eigen Λ dan vektor eigen Φ untuk masing-masing matriks kovarian diatas.

$$\Phi_1 = \begin{bmatrix} -0.79948926 & 0.600680375 \\ 0.600680375 & 0.7994892663 \end{bmatrix} \quad \Phi_2 = \begin{bmatrix} -0.87171048798 & -0.490021249671 \\ -0.490021249671 & 0.8717104879 \end{bmatrix}$$

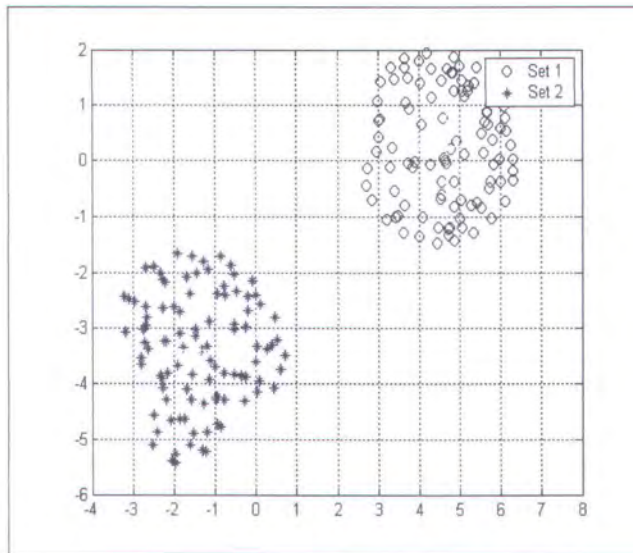
$$\Lambda_1 = \begin{bmatrix} 0.35981207 & 0 \\ 0 & 2.25321436 \end{bmatrix} \quad \Lambda_2 = \begin{bmatrix} 0.31077369 & 0 \\ 0 & 0.64712771 \end{bmatrix}$$

Dengan menggunakan rumus Transformasi Whitening,

Matriks Transformasi Whitening untuk masing-masing himpunan adalah sebagai berikut

$$T_1 = \begin{bmatrix} -1.3328300372 & 1.0013953664 \\ 0.4001678444 & 0.5326125335 \end{bmatrix} \quad T_2 = \begin{bmatrix} -1.5636881616 & -0.8790079241 \\ -0.6091439185 & 1.0836206447 \end{bmatrix}$$

Hasil Transformasi Whitening terhadap data asal digambarkan pada diagram berikut. Matriks Kovarian pada masing-masing data hasil transformasi sama dengan matriks identitas.



Gambar 2. 2. Plot data hasil PCA dan transformasi *whitening*

Dengan matrik masukan A maka untuk mencari matrik baru hasil whitening menggunakan rumus:

$$\tilde{A} = W_h A \quad (2.18)$$

Matrik kovarian dari matrik masukan adalah $A A^T$, misalkan diambil contoh sebuah korelasi masukan matrik $R_{AA} = A A^T$ dan menemukan nilai eigen dan vektor eigen (R_{AA} adalah matrik simetris)

$$R_{AA} = P \Lambda P^T \quad (2.19)$$

dimana P adalah matrik vektor eigen dan $\Lambda = \text{diag}\{\lambda_1, \lambda_2, \dots, \lambda_k\}$ matrik diagonal dari nilai eigen. Dari persamaan 2.13 maka:

$$\begin{aligned} R_{AA} &= P \Lambda P^T = P \Lambda^{1/2} \Lambda^{1/2} P^T \\ &= A A^T = \{W h^{-1} \tilde{x}\} \{\tilde{x}^T W h^{-T}\} = W h^{-1} \{\tilde{x} \tilde{x}^T\} W h^{-T} = W h^{-1} W h^{-T} \end{aligned} \quad (2.20)$$

dari persamaan 2.12, maka rumus Transformasi Whitening adalah:

$$W_h = (P \Lambda^{1/2})^{-1}$$

$$\begin{aligned}
&= \Lambda^{-1/2} P^{-1} \\
&= \Lambda^{-1/2} P^T
\end{aligned} \tag{2.21}$$

Sehingga untuk mencari matrik baru hasil transformasi whitening V , berdasarkan persamaan 2.11 dan persamaan 2.15 adalah dengan rumusan

$$V = \Lambda^{-1/2} P^T A \tag{2.22}$$

dengan $\Lambda^{-1/2} = \text{diag}\{\lambda_1^{-1/2}, \lambda_2^{-1/2}, \dots, \lambda_k^{-1/2}\}$, dan $\Lambda^{-1/2} \in \mathbb{R}^{m \times m}$.

2.2 ANALISA PAKET WAVELET

Dalam sistem pengenalan wajah ini digunakan dua fitur ekstraktor untuk mengekstraksi fitur global dan fitur lokal. PCA digunakan sebagai pengeksraksi fitur global untuk mendapatkan basis proyeksi yang optimal dan untuk lebih meningkatkan jangkauan ruang solusi maka PCA dimodifikasi dengan transformasi *whitening*. Sedangkan untuk mengekstraksi fitur lokalnya digunakan metode Wavelet yang akan mendekomposisi citra wajah sehingga dapat memberikan info lokal wajah.

Wavelet adalah fungsi yang memenuhi persyaratan matematika tertentu yang mampu melakukan dekomposisi terhadap sebuah fungsi secara terhirarki untuk digunakan merepresentasikan data atau fungsi lainnya.

Transformasi wavelet merupakan sebuah fungsi konversi yang dapat membagi fungsi atau sinyal ke dalam komponen frekuensi atau skala yang berbeda. Lebih khusus lagi untuk pengklasifikasian fitur dapat digunakan Transformasi Paket Wavelet (TPW). Dengan menerapkan TPW pada sebuah citra akan diperoleh

koefisien-koefisien paket wavelet yang unik untuk setiap cuplikan citra tersebut. Koefisien-koefisien tersebut menunjukkan karakteristik yang unik dari sinyal pada kanal frekuensi yang berbeda-beda seperti dalam analisa multiresolusi dan menyajikan suatu informasi yang unik untuk setiap fitur. Karakteristik inilah yang akan digunakan untuk mengklasifikasikan fitur.

2.2.1 Transformasi Wavelet

Transformasi wavelet (TW) adalah suatu metoda pendekatan fungsi dengan lokalisasi *ruang frekuensi*. TW dapat menganalisa sinyal menurut ruang dengan menggunakan fungsi-fungsi dasar, yaitu fungsi induk wavelet. Fungsi-fungsi dasar tersebut diperlukan sebagai suatu “*jendela*” pendekatan (aproksimasi) yang terlokalisir pada interval ruang tertentu. Fungsi induk wavelet didefinisikan sebagai berikut:

$$\Psi_{a,b}(x) = |a|^{-\frac{1}{2}} \Psi \left[\frac{x-b}{a} \right] \quad (2.23)$$

dimana: $\Psi_{a,b}(x)$ adalah fungsi induk wavelet, a adalah parameter dilatasi ($a \in \text{Real}$), b adalah parameter tranlasi ($b \in \text{Real}$). Parameter a menunjukkan lebar dari kurva wavelet, parameter b menunjukkan bahwa lokalisasi kurva wavelet terpusat pada interval raung $x = b$. Untuk data diskrit (seperti citra) diperlukan Transformasi Wavelet Diskrit (TWD), sehingga persamaan 2.17 menjadi:

$$\Psi_{j,k}(x) = 2^{-\frac{j}{2}} \Psi(2^j x - k) \quad (2.24)$$

Untuk mencapai lokalisasi ruang frekuensi, suatu fungsi harus diekspansi dalam bentuk basis orthonormal wavelet. Oleh karena itu, selain fungsi wavelet $\Psi(x)$ juga diperlukan fungsi skala $\phi(x)$ yang dalam bentuk diskritnya didefinisikan

sebagai:

$$\phi_{j,k}(x) = 2^{\frac{1}{2}} \phi(2^j x - k) \quad (2.25)$$

dimana: $\phi_{j,k}(x)$ adalah scaling filter, j adalah parameter dilatasi, k adalah parameter translasi. Parameter j menunjukkan lebar dari kurva wavelet, parameter k menunjukkan bahwa lokalisasi kurva wavelet terpusat pada interval ruang $x = k$.

2.1.1 Analisa Multiresolusi

Gabungan fungsi wavelet $\psi(x)$ dan fungsi skala $\phi(x)$ dapat membentuk sekumpulan basis, sehingga suatu sinyal atau fungsi $f(x)$ dapat diekspresikan sebagai kombinasi linier dari keduanya, yaitu:

$$f(x) = \sum_{j,k} c_{j,k} \psi_{j,k}(x) + \sum_{j,k} d_{j,k} \phi_{j,k}(x) \quad (2.26)$$

Koefisien detail $c_{j,k}$ merepresentasikan citra detail pada level j sebagai hasil pemfilteran *high_pass*, sedangkan koefisien *smooth* $d_{j,k}$ merepresentasikan citra *smooth* pada level j sebagai hasil pemfilteran *low_pass*. Koefisien-koefisien dalam sistem wavelet disebut sebagai Transformasi Wavelet Diskrit dari sinyal $f(x)$. Terlihat dari persamaan 2.26 bahwa sinyal atau fungsi asal dipisahkan menjadi komponen frekuensi tinggi (suku pertama) dan komponen frekuensi rendah (suku kedua). Dengan cara ini pengaplikasian wavelet akan menghasilkan resolusi yang berbeda-beda untuk suatu fungsi. Hal inilah yang dinamakan analisa multiresolusi.

Suatu ruang V_1 dapat didekomposisi menjadi 2 sub ruang V_0 dan W_0 , kemudian subruang V_0 yang diperoleh, didekomposisi lagi menjadi 2 sub ruang V_{-1} dan W_{-1} ,

demikian seterusnya. Secara matematis, hal ini dapat digambarkan sebagai urutan naik dari sub-sub ruang tertutup V_j dari $L^2(\mathbb{R})$ dan dapat dituliskan sebagai:

$$V_{-2} \subset V_{-1} \subset V_0 \subset V_1 \dots L^2(\mathbb{R}) \quad (2.27)$$

Fungsi skala $\phi(x)$ adalah suatu fungsi yang apabila didilasikan dan ditranslasikan, membentuk basis orthonormal dari sub-sub ruang V_j tersebut. Sedangkan fungsi wavelet $\psi(x)$ adalah fungsi induk yang translasi dan dilasinya didefinisikan untuk menjadi basis orthonormal dari sub-sub ruang W_j , dimana W_j adalah komplemen orthogonal dari V_j untuk membentuk ruang V_{j+1} .

$$V_{j+1} = V_j \oplus W_j \quad (2.28)$$

2.1.2 Konvolusi

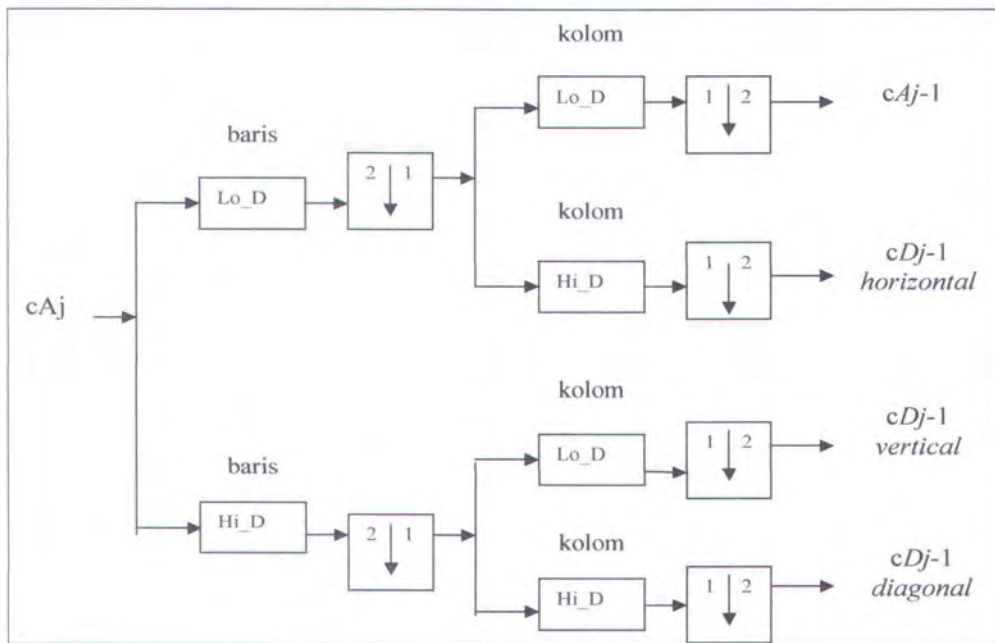
Konvolusi adalah operasi matematika sederhana yang fundamental untuk beberapa operator pemrosesan citra. Hal ini digunakan dalam pemrosesan citra untuk mengimplementasikan operator yang nilai pixel keluarannya adalah kombinasi linear sederhana dari nilai pixel keluarannya.

Konvolusi dengan filter low-pass menghasilkan pendekatan citra dan konvolusi dengan filter high-pass dengan arah horizontal, vertical dan diagonal menghasilkan 3 citra detail.

2.1.3 Dekomposisi

Dekomposisi adalah suatu proses yang memecah citra, dalam hal ini fitur lokal pada wajah, secara baris dan secara kolom menjadi empat bagian dengan menggunakan nilai-nilai wavelet yang ada, sehingga masukan awal dari proses

dekomposisi ini adalah nilai masing-masing piksel dari keluaran fitur. Proses dekomposisi citra menggunakan 2 fungsi konvolusi, yaitu fungsi wavelet (high-pass filter) dan fungsi skala (low-pass filter). Fungsi Wavelet menghasilkan nilai koefisien detail dan fungsi skala menghasilkan nilai koefisien rata-rata. Gambar dibawah ini menunjukkan proses dekomposisi secara rinci:



Gambar 2. 3. Skema Dekomposisi

Keterangan:

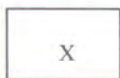


Index kolom yang genap dihilangkan



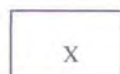
Index baris yang genap dihilangkan

baris



Konvolusi dengan filter X pada bagian barisnya

kolom



Konvolusi dengan filter X pada bagian kolomnya

Pada tugas akhir ini, transformasi yang digunakan adalah Transformasi Haar Wavelet, yang mempunyai koefisien filter:

1. Low-pass filter adalah 0,707106781187 dan 0,707106781187
2. High-pass filter adalah 0,707106781187 dan -0,707106781187

Persamaan dari Transformasi Haar Wavelet adalah:

1. Fungsi skala (low-pass filter) dihitung dengan persamaan:

$$a[i] = h[0]s[2i] + h[1]s[2i+1] \quad (2.29)$$

2. Fungsi wavelet (high-pass filter) dihitung dengan persamaan:

$$c[i] = g[0]s[2i] + g[1]s[2i+1] \quad (2.30)$$

Keterangan:

$i = 0, 1, 2, \dots, n$

$a_{1,2,\dots,n}$ dan $c_{1,2,\dots,n}$ = vektor keluaran

$h_{1,2,\dots,n}$ dan $g_{1,2,\dots,n}$ = vektor filter koefisien Haar (skala dan wavelet)

$s_{1,2,\dots,n}$ = vektor masukan

dari persamaan 2.29 dan 2.30, jika di dibuat dalam bentuk matrik adalah:

$$\begin{pmatrix} h[0] & h[1] & 0 & 0 & 0 & 0 & 0 & 0 \\ g[0] & g[1] & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & h[0] & h[1] & 0 & 0 & 0 & 0 \\ 0 & 0 & g[0] & g[1] & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & h[0] & h[1] & 0 & 0 \\ 0 & 0 & 0 & 0 & g[0] & g[1] & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & h[0] & h[1] \\ 0 & 0 & 0 & 0 & 0 & 0 & g[0] & g[1] \end{pmatrix} \times \begin{pmatrix} s0 \\ s1 \\ s2 \\ s3 \\ s4 \\ s5 \\ s6 \\ s7 \end{pmatrix} = \begin{pmatrix} a1 \\ c1 \\ a2 \\ c2 \\ a3 \\ c3 \\ a4 \\ c4 \end{pmatrix}$$

Misal terdapat data :

$$\begin{pmatrix} 47 & 73 & 47 & 30 & 45 & 37 & 27 & 8 \\ 67 & 69 & 46 & 42 & 63 & 26 & 24 & 14 \\ 72 & 40 & 59 & 36 & 62 & 11 & 15 & 57 \\ 67 & 35 & 72 & 52 & 51 & 35 & 30 & 83 \\ 39 & 37 & 65 & 61 & 41 & 20 & 24 & 70 \\ 51 & 70 & 54 & 68 & 37 & 36 & 111 & 88 \\ 64 & 50 & 44 & 50 & 44 & 48 & 119 & 148 \\ 86 & 46 & 50 & 37 & 34 & 86 & 99 & 145 \end{pmatrix}$$

maka untuk mendekomposisinya adalah mengalikan filter dengan data. Contoh untuk konvolusi terhadap KOLOM adalah seperti dibawah ini, misalnya pada kolom pertama, yaitu:

$$\begin{pmatrix} \frac{1}{2} & \frac{1}{2} & 0 & 0 & 0 & 0 & 0 & 0 \\ \frac{1}{2} & -\frac{1}{2} & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & \frac{1}{2} & \frac{1}{2} & 0 & 0 & 0 & 0 \\ 0 & 0 & \frac{1}{2} & -\frac{1}{2} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \frac{1}{2} & \frac{1}{2} & 0 & 0 \\ 0 & 0 & 0 & 0 & \frac{1}{2} & -\frac{1}{2} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & \frac{1}{2} & \frac{1}{2} \\ 0 & 0 & 0 & 0 & 0 & 0 & \frac{1}{2} & -\frac{1}{2} \end{pmatrix} * \begin{pmatrix} 47 \\ 67 \\ 72 \\ 67 \\ 39 \\ 51 \\ 64 \\ 86 \end{pmatrix} = \begin{pmatrix} 7 \\ -10 \\ 9 \\ 2.5 \\ 45 \\ -6 \\ 75 \\ -11 \end{pmatrix}$$

Jika kolom ke dua sampai ke tujuh dilakukan operasi yang sama dengan contoh diatas, maka hasil dekomposisinya matriknya adalah

$$\begin{pmatrix} 57 & 71 & 46.5 & 36 & 54 & 31.5 & 25.5 & 11 \\ -10 & 2 & 0.5 & -6 & -9 & 5.5 & 1.5 & -3 \\ 69.5 & 37.5 & 65.5 & 44 & 56.5 & 23 & 22.5 & 70 \\ 2.5 & 2.5 & -6.5 & -8 & 5.5 & -12 & -7.5 & -13 \\ 45 & 53.5 & 59.5 & 64.5 & 39 & 28 & 67.5 & 79 \\ -6 & -16.5 & 5.5 & -3.5 & 2 & -8 & -43.5 & -9 \\ 75 & 48 & 47 & 43.5 & 39 & 67 & 109 & 146.5 \\ -11 & 2 & -3 & 6.5 & 5 & -19 & 10 & 1.5 \end{pmatrix}$$



Dibentuk suatu matrik baru hasil penggabungan **angka yang tebal** dengan **angka yang tebal** dari matrik diatas dan akan dihasilkan matrik hasil konvolusi dengan filter koefisien low-pass

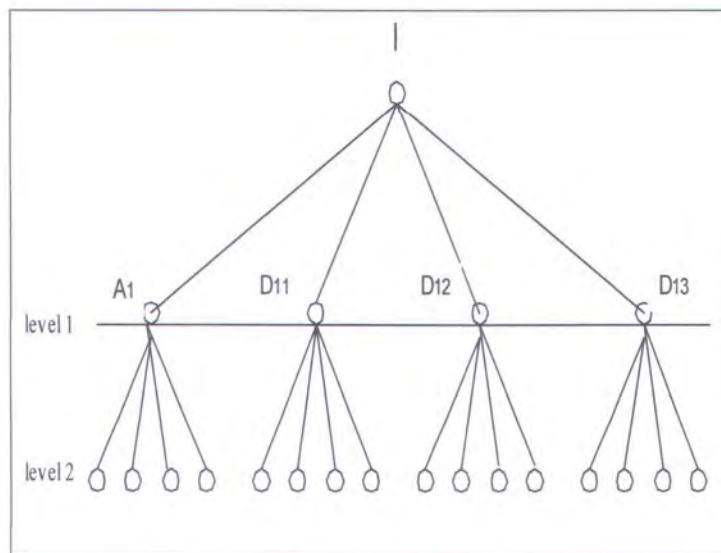
$$\begin{pmatrix} 57 & 71 & 46.5 & 36 & 54 & 31.5 & 25.5 & 11 \\ 69.5 & 37.5 & 65.5 & 44 & 56.5 & 23 & 22.5 & 70 \\ 45 & 53.5 & 59.5 & 64.5 & 39 & 28 & 67.5 & 79 \\ 75 & 48 & 47 & 43.5 & 39 & 67 & 109 & 146.5 \end{pmatrix}$$

Dan hasil hasil penggabungan angka biasa dengan angka biasa akan dihasilkan matrik hasil konvolusi dengan filter koefisien high-pass

$$\begin{pmatrix} -10 & 2 & 0.5 & -6 & -9 & 5.5 & 1.5 & -3 \\ 2.5 & 2.5 & -6.5 & -8 & 5.5 & -12 & -7.5 & -13 \\ -6 & -16.5 & 5.5 & -3.5 & 2 & -8 & -43.5 & -9 \\ -11 & 2 & -3 & 6.5 & 5 & -19 & 10 & 1.5 \end{pmatrix}$$

Terdapat Dua Macam Metode Dekomposisi Paket Wavelet, yaitu Paket Wavelet Baku dan Paket Wavelet Lengkap. Dalam tugas akhir ini digunakan Paket Wavelet Lengkap karena pengambilan sub citra pada level paling bawah akan menghasilkan informasi yang berarti [10].

Pada dekomposisi Paket Wavelet Lengkap, citra dibagi menjadi pendekatan citra dan detail citra. Kemudian pendekatan citra dibagi lagi kedalam pendekatan citra dan detail citra level kedua. Begitu juga dengan detail citra level pertama dibagi seperti pendekatan citranya. Sehingga pada dekomposisi level kedua akan didapatkan satu pendekatan citra (citra beresolusi rendah) dan 15 detail citra.

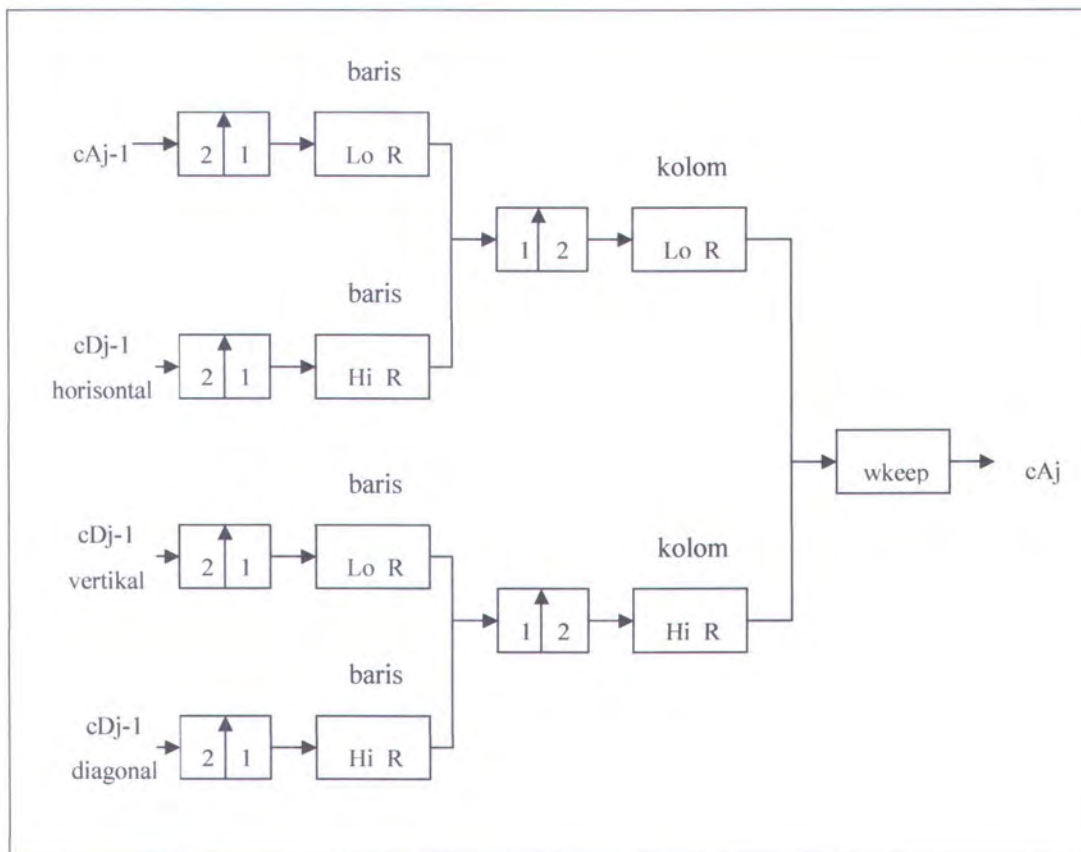


Gambar 2. 4. Bagan Paket Wavelet Complete

Pada tugas akhir ini, Paket Wavelet yang digunakan adalah dekomposisi level 2. Hal ini dilakukan agar *encode* dari sinyal dapat lebih bagus. Dalam hal ini tidak diperlukan dekomposisi pada level yang lebih rendah lagi, karena setelah dekomposisi level kedua, citra wajah akan sangat kecil dan tidak ada informasi tambahan yang akan didapat [10].

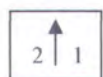
2.2.5 Rekonstruksi

Rekonstruksi merupakan kebalikan dari dekomposisi, yaitu suatu proses yang menggabungkan sub-sub citra wajah ke citra wajah semula. Proses ini juga melibatkan nilai-nilai filter orthogonal yang merupakan invers dari proses dekomposisi. Invers filter yang digunakan adalah LoF_R (Low Pass Filter Rekonstruksi) dan HiF_R (High Pass Filter Rekonstruksi). Gambar dibawah ini menunjukkan proses rekonstruksi secara rinci:

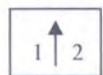


Gambar 2. 5. Skema Rekonstruksi

Keterangan:

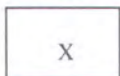


Sisipkan



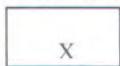
Sisipkan

baris

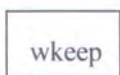


Konvolusi dengan filter X pada bagian barisnya

kolom



Konvolusi dengan filter X pada bagian kolomnya



wkeep

Misal matrik filter rekonstruksi L^* dan H^* , adalah:

$$L^* = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix} \quad H^* = \begin{pmatrix} 1 & 0 & 0 & 0 \\ -1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & -1 & 0 & 0 \end{pmatrix}$$

Untuk setiap baris hasil dekomposisi (matriks Pendekatan), kalikan matriks L^* dan H^* tersebut dengan bagian dari matrik detil.

Contohnya baris pertama matriks pendekatan dan matriks detil horisontal pada contoh hasil dekomposisi:

$$\begin{pmatrix} 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 \end{pmatrix} * \begin{pmatrix} 57 \\ 59.5 \\ 45 \\ 75 \end{pmatrix} + \begin{pmatrix} 1 & 0 & 0 & 0 \\ -1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & -1 \end{pmatrix} * \begin{pmatrix} -10 \\ 2.5 \\ -6 \\ -11 \end{pmatrix}$$

$$= \begin{pmatrix} 57 \\ 52 \\ 59.5 \\ 59.5 \\ 45 \\ 45 \\ 75 \\ 75 \end{pmatrix} + \begin{pmatrix} -10 \\ 10 \\ 2.5 \\ -2.5 \\ -6 \\ 6 \\ -11 \\ 11 \end{pmatrix} = \begin{pmatrix} 47 \\ 67 \\ 72 \\ 67 \\ 39 \\ 51 \\ 64 \\ 86 \end{pmatrix}$$

Jika dilakukan terhadap kolom-kolom selanjutnya maka akan hasil rekonstruksi

sama dengan data masukan sebelum didekomposisi, sehingga dapat dikatakan proses dekomposisi yang dilakukan benar.

47	73	47	30	45	37	27	8
67	69	46	42	63	26	24	14
72	40	59	36	62	11	15	57
67	35	72	52	51	35	30	83
39	37	65	61	41	20	24	70
51	70	54	68	37	36	111	88
64	50	44	50	44	48	119	148
86	46	50	37	34	86	99	145

2.3 Sistem Inferensi Fuzzy

Sistem Inferensi Fuzzy diketahui sebagai sistem berbasis aturan fuzzy, model fuzzy, *Fuzzy associative memories* (FAM) atau *fuzzy controllers* ketika digunakan sebagai pengontrol. Secara mendasar sistem inferensi fuzzy disusun dari 5 blok fungsional:

1. Aturan dasar (*Rule base*), berisi *fuzzy rule if-then*
2. Basis Data, yang mendefinisikan fungsi keanggotaan dari kumpulan fuzzy yang digunakan dalam *fuzzy rule*
3. Unit pembuat keputusan, yang membentuk operasi inferen dalam rule
4. Antarmuka *fuzzification*, yang mentransform masukancrisp ke derajat yang cocok dengan nilai linguistic
5. Antarmuka *defuzzification*, yang mentransform hasil fuzzy dari inferen ke keluaran berbentuk crisp

Langkah-langkah pembentukan *fuzzy reasoning* (operasi inferen berdasarkan

aturan fuzzy *if-then*) oleh Sistem Inferensi Fuzzy adalah:

1. Membandingkan variabel masukan dengan fungsi keanggotaan di bagian *premis* untuk mendapatkan nilai keanggotaan (ukuran komparabilitas) untuk tiap label linguistic.
2. Mengkombinasikan (dengan operator T-norm, biasanya dikali atau min) nilai keanggotaan di bagian *premis* untuk mendapatkan *firing strength* (bobot) tiap rule
3. Men-*generate consequent* dari tiap rule tergantung pada *firing strength*
4. Meng-agregasi *consequent* yang berkualitas untuk menghasilkan keluaran crisp.

2.3.1.1 Metode Sistem Inferensi Fuzzy

Ada 3 metode yang digunakan dalam melakukan Sistem Inferensi Fuzzy, yaitu; max-min, additive dan probabilistik OR (probor)

1. Metode Max (maximum)

Pada metode ini, solusi himpunan fuzzy diperoleh dengan cara mengambil nilai maksimum aturan, kemudian menggunakannya untuk memodifikasi daerah fuzzy, dan mengaplikasikannya ke keluaran dengan menggunakan operator OR (union). Jika semua proposisi telah dievaluasi maka keluaran akan berisi suatu himpunan fuzzy yang merefleksikan kontribusi dari tiap-tiap proposisi. Secara umum dapat dituliskan:

$$\mu_{sf}[x_i] \leftarrow \max (\mu_{sf}[x_i], \mu_{kf}[x_i]) \quad (2.39)$$

dengan:

$\mu_{sf}[x_i]$ = nilai keanggotaan solusi fuzzy sampai aturan ke-i;

$\mu_{kf}[x_i]$ = nilai keanggotaan konsekuensi fuzzy aturan ke-i;

Apabila digunakan fungsi implikasi MIN, maka metode komposisi ini sering disebut dengan nama MAX-MIN atau MIN-MAX atau MAMDANI

2. Metode Additive (Sum)

Pada metode ini, solusi himpunan fuzzy diperoleh dengan cara melakukan *bounded-sum* terhadap semua keluaran daerah fuzzy. Secara umum dituliskan:

$$\mu_{sf}[x_i] \longleftarrow \min(1, \mu_{sf}[x_i] + \mu_{kf}[x_i]) \quad (2.40)$$

dengan:

$\mu_{sf}[x_i]$ = nilai keanggotaan solusi fuzzy sampai aturan ke-i;

$\mu_{kf}[x_i]$ = nilai keanggotaan konsekuensi fuzzy aturan ke-i;

3. Metode Probabilistik OR (probor)

Pada metode ini, solusi himpunan fuzzy diperoleh dengan cara melakukan *product* terhadap semua keluaran daerah fuzzy. Secara umum dituliskan:

$$\mu_{sf}[x_i] \longleftarrow (\mu_{sf}[x_i] + \mu_{kf}[x_i]) - (\mu_{sf}[x_i] * \mu_{kf}[x_i]) \quad (2.41)$$

dengan:

$\mu_{sf}[x_i]$ = nilai keanggotaan solusi fuzzy sampai aturan ke-i;

$\mu_{kf}[x_i]$ = nilai keanggotaan konsekuensi fuzzy aturan ke-i;

2.3.1.2 Defuzzifikasi

Masukan dari proses defuzzifikasi adalah suatu himpunan fuzzy yang diperoleh dari komposisi aturan-aturan fuzzy, sedangkan keluaran yang dihasilkan merupakan suatu bilangan pada domain himpunan fuzzy tersebut. Sehingga jika diberikan suatu himpunan fuzzy dalam range tertentu, maka harus dapat diambil suatu nilai crisp tertentu sebagai keluaran.

Metode-metode defuzzifikasi adalah:

1. Metode Centroid (*Composite Moment*)

Pada metode ini, solusi crisp diperoleh dengan cara mengambil titik pusat daerah fuzzy. Secara umum dirumuskan:

$$z = \frac{\int z \mu(z) dz}{\int \mu(z) dz} \quad \text{atau} \quad z = \frac{\sum_{j=1}^n z_j \mu(z_j)}{\sum_{j=1}^n \mu(z_j)} \quad (2.42)$$

Ada 2 keuntungan menggunakan metode centroid, yaitu:

- a. Nilai defuzzy akan bergerak secara halus sehingga perubahan dari suatu topologi himpunan fuzzy ke topologi berikutnya juga akan berjalan dengan halus
- b. Mudah dihitung

2. Metode Bisektor

Pada metode ini, solusi crisp diperoleh dengan cara mengambil nilai pada domain fuzzy yang memiliki nilai keanggotaan separo dari jumlah total nilai keanggotaan pada daerah fuzzy. Secara umum dituliskan:

$$z_p \text{ sedemikian hingga } \int_{R_l}^p \mu(z) dz = \int_p^{R_u} \mu(z) dz \quad (2.43)$$

3. Metode Mean of Maximum (MOM)

Pada metode ini, solusi crisp diperoleh dengan cara mengambil nilai rata-rata domain yang memiliki nilai keanggotaan maksimum.

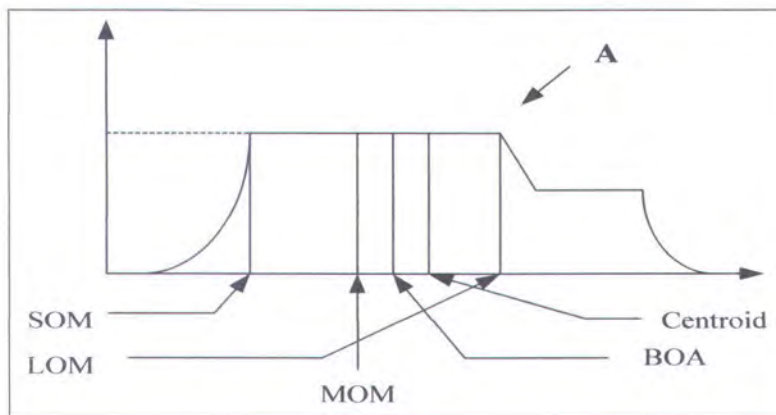
4. Metode Largest of Maximum (LOM)

Pada metode ini, solusi crisp diperoleh dengan cara mengambil nilai terbesar dari

domain yang memiliki nilai keanggotaan maksimum.

5. Metode Smallest of Maximum (SOM)

Pada metode ini, solusi crisp diperoleh dengan cara mengambil nilai terkecil dari domain yang memiliki nilai keanggotaan maksimum.



Gambar 2. 6. Skema Defuzzifikasi untuk mendapatkan nilai keluaran

2.3.1.3 Penalaran Fuzzy Metode SUGENO

Pada metode MAMDANI, baik masukan(antiseden) maupun keluaran (konsekuen) sistem berupa himpunan fuzzy. Penalaran dengan metode SUGENO hampir sama dengan penalaran MAMDANI, hanya saja keluaran (konsekuen) sistem tidak berupa himpunan fuzzy, melainkan berupa konstanta atau persamaan linier.

- Model Fuzzy SUGENO Orde-Nol

Secara umum bentuk model fuzzy SUGENO Orde-Nol adalah:

$$\text{If } (x_1 \text{ is } A_1) \cdot (x_2 \text{ is } A_2) \cdot \dots \cdot (x_N \text{ is } A_N) \text{ then } z = k \quad (2.44)$$

Dengan A_i adalah himpunan fuzzy ke-I sebagai anteseden dan k adalah suatu konstanta (tegas) sebagai konsekuen

- Model Fuzzy SUGENO Orde-Satu

Secara umum bentuk model fuzzy SUGENO Orde-Satu adalah:

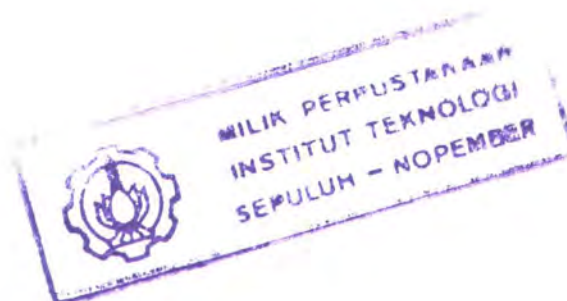
$$\text{If } (x_1 \text{ is } A_1) \cdot (x_2 \text{ is } A_2) \cdot \dots \cdot (x_N \text{ is } A_N) \text{ then } z = p_1 * x_1 + \dots + p_N * x_N + q$$

Dengan A_i adalah himpunan fuzzy ke- i sebagai anteseden dan k adalah suatu konstanta (tegas) sebagai konsekuen. Apabila komposisi aturan menggunakan metode SUGENO, maka defuzzifikasi dilakukan dengan cara mencari nilai rata-ratanya.

2.3.2 Jaringan Adaptif

Pembelajaran dari jaringan adaptif dalam kenyataannya adalah sebuah *superset* dari semua macam *feedforward* neural network dengan kemampuan pembelajaran *supervised*. Jaringan adaptif adalah struktur jaringan yang terdiri dari simpul (node) dan rantai yang menghubungkan simpul-simpul secara langsung. Selain itu, sebagian atau semua node adalah adaptif, yang berarti setiap keluaran dari simpul-simpul ini tergantung pada parameter yang ada di simpul dan aturan pembelajaran menspesifikasikan bagaimana parameter-parameter ini seharusnya diubah untuk meminimalisasi pengukuran error.

Dasar pembelajaran aturan dari jaringan adaptif adalah berbasis pada turunan gradient (*gradient descent*) dan rangkaian aturan (*rule chain*). Karena dasar pembelajaran aturan berbasis metode gradient terkenal dengan kelambatannya dan cenderung terjebak dalam *local minima*, sehingga di ajukan pembelajaran *hybrid* yang dapat meningkatkan kecepatan proses pembelajaran.



2.3.2.1 Arsitektur Jaringan Adaptif

Sesuai dengan namanya, jaringan adaptif adalah struktur jaringan yang jalannya masukan dan keluaran secara keseluruhan ditentukan oleh kumpulan parameter yang dapat dirubah. Konfigurasi dari jaringan adaptif terdiri dari kumpulan simpul yang terhubung secara langsung melalui rantai (*links*), dimana setiap simpul melakukan fungsi simpul secara statis ketika ada sinyal yang masuk untuk menghasilkan satu simpul keluaran, dan untuk setiap rantai menetapkan arah aliran sinyal dari simpul satu ke simpul lainnya. Biasanya fungsi simpul adalah fungsi parameter, dimana parameter-parameter tersebut dapat dirubah. Dengan merubah parameter ini, fungsi simpul akan berubah dan jalan keseluruhan dari jaringan adaptif juga akan berubah sebaik mungkin.

Setiap simpul dalam jaringan adaptif membentuk pemetaan statis dari masukan ke keluarannya. Dapat dikatakan, setiap keluaran hanya tergantung pada yang ada, disini tidak ada state dinamis atau state internal di dalam simpul. Umumnya, jaringan adaptif adalah bermacam-macam dan tiap simpul memungkinkan untuk mempunyai fungsi simpul khusus yang berbeda dengan simpul yang lain. Rantai dalam jaringan adaptif hanya digunakan untuk menentukan arah perambatan simpul keluaran, umumnya tidak ada menghubungkan bobot atau parameter dengan rantai.

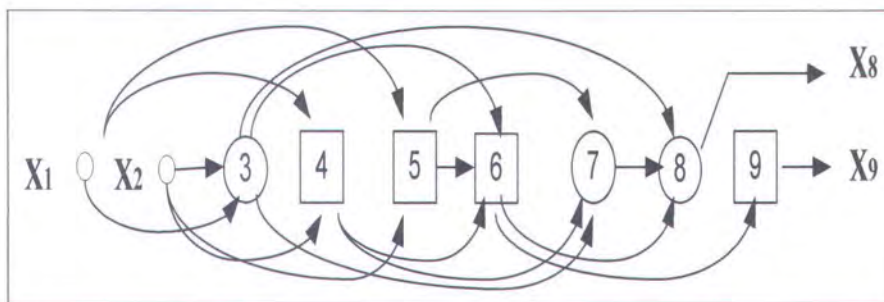
Parameter-parameter dalam jaringan adaptif disalurkan ke dalam simpul-simpul, sehingga setiap node mempunyai kumpulan parameter lokal. Simpul adaptif direpresentasikan dalam bentuk kotak, dimana kumpulan parameter simpulnya tidak kosong, dan fungsi simpul tergantung pada nilai parameter. Tetapi jika

simpul tidak mempunyai kumpulan parameter, dan fungsi simpulnya tetap (*fix*) digunakan simpul berbentuk bundar untuk merepresentasikannya, dan simpul ini dinamakan simpul tetap (*fix node*).

Jaringan adaptif umumnya diklasifikasikan ke dalam dua kategori berdasarkan tipe hubungannya, yaitu:

1. *Feedforward*

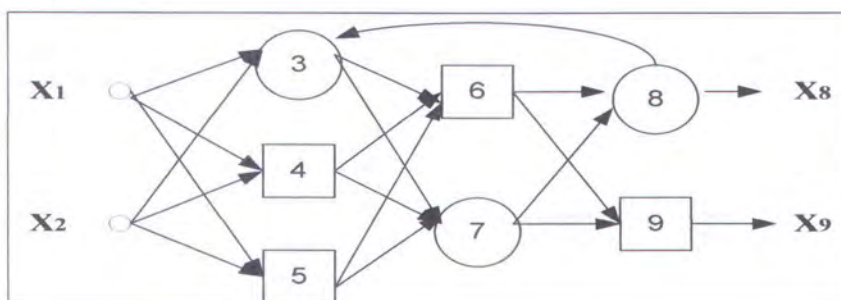
Jaringan adaptif yang keluaran setiap simpulnya merambat dari sebelah kiri (keluaran) ke sebelah kanan (keluaran) .



Gambar 2. 7. Jaringan adaptif Feedforward

2. *Recurrent*

Jaringan adaptif dimana ada rantai yang kembali sehingga membentuk *circular path* dalam jaringan



Gambar 2. 8. Jaringan adaptif Recurent

Abu Hurairah r.a berkata: Rasulullah SAW. Bersabda: Jika mati anak Adam (manusia) maka terputuslah amal usahanya sendiri kecuali tiga: sedekah yang berjalan terus. Ilmu pengetahuan yang berguna. Anak yang sholeh yang selalu mendoakan padanya – Muslim

BAB III

PENERAPAN ANFIS PADA

PENGENALAN WAJAH

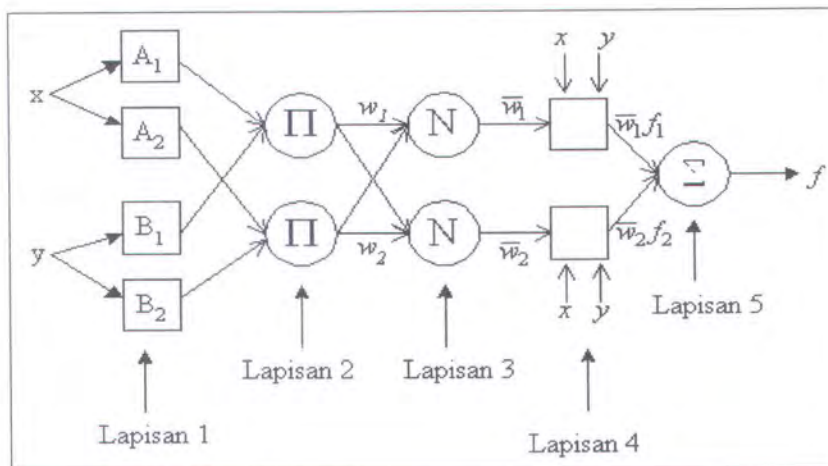
BAB III

PENERAPAN ANFIS PADA PENGENALAN WAJAH

Pada bab ini akan dijelaskan mengenai penerapan metode *Adaptive Neuro-Fuzzy Inference System* (ANFIS) sebagai mesin pembelajaran pada sistem pengenalan wajah, yang mempunyai metode pembelajaran *hybrid* yaitu pembelajaran arah maju (*forward pass*) dengan menerapkan metode Galat Kuadrat Terkecil (*Recursive Least Square Estimator*- RLSE) dan arah mundur (*backward pass*) yang menggunakan metode turunan gradient (*Gradient Descent*). Pembelajaran *hybrid* ini digunakan untuk memperbaharui parameter-parameter premis dan konsekuen.

3.1 STRUKTUR ANFIS

ANFIS adalah metode yang merupakan penggabungan mekanisme Sistem Inferensi Fuzzy yang digambarkan dalam arsitektur jaringan syaraf [7]. Sistem Inferensi Fuzzy yang digunakan adalah Sistem Inferensi Fuzzy model Tagaki-Sugeno-Kang (TSK) Orde-Satu dengan pertimbangan kesederhanaan dan kemudahan komputasi. Struktur ANFIS yang dipakai dalam tugas akhir ini pada gambar 3.1. Pada gambar 3.1, struktur ANFIS terdiri atas lima lapisan dengan fungsi yang berbeda untuk tiap lapisannya. Lambang kotak menyatakan simpul adaptif artinya nilai parameternya bisa berubah dengan pembelajaran dan lambang lingkaran menyatakan simpul nonadaptif yang nilainya tetap.



Gambar 3.1. Struktur ANFIS

Penjelasan tentang lapisan-lapisan yang terdapat pada struktur ANFIS adalah:

1. Lapisan 1

Semua simpul pada lapisan ini adalah simpul adaptif (parameter dapat berubah) dengan fungsi simpul :

$$\begin{aligned} O_{1,i} &= \mu_{A_i}(x), \quad \text{untuk } i = 1, 2, \text{ atau} \\ O_{1,i} &= \mu_{B_{i-2}}(y), \quad \text{untuk } i = 3, 4 \end{aligned} \quad (3.1)$$

dengan x dan y adalah masukan pada simpul i , A_i (atau B_{i-2}) adalah fungsi keanggotaan masing-masing simpul. Simpul $O_{1,i}$ berfungsi untuk menyatakan derajat keanggotaan tiap masukan terhadap himpunan fuzzy A dan B . Fungsi keanggotaan yang dipakai dalam tugas akhir ini Fungsi Gauss2mf yang merupakan kombinasi antara 2 kurva. Parameter sig1 , $c1$, sig2 dan $c2$ pada fungsi keanggotaan gauss2mf dinamakan parameter premis yang adaptif.

2. Lapisan 2

Semua simpul pada lapisan ini adalah nonadaptif (parameter tetap). Fungsi simpul ini adalah mengalikan setiap sinyal masukan yang datang. Fungsi simpul :

$$O_{2,i} = w_i = \mu_{A_i}(x) \cdot \mu_{B_i}(y), \quad i = 1, 2 \quad (3.2)$$

Tiap keluaran simpul menyatakan derajat pengaktifan (*firing strength*) tiap aturan fuzzy. Banyaknya simpul pada lapisan ini menunjukkan banyaknya aturan yang dibentuk. Fungsi perkalian yang digunakan adalah interpretasi kata hubung **AND** dengan menggunakan operator t-norm.

3. Lapisan 3

Setiap simpul pada lapisan ini adalah simpul non adaptif yang menampilkan fungsi derajat pengaktifan ternormalisasi (*normalized firing strength*) yaitu rasio keluaran simpul ke- i pada lapisan sebelumnya terhadap seluruh keluaran lapisan sebelumnya, dengan bentuk fungsi simpul:

$$O_{3,i} = \bar{w}_i = \frac{w_i}{w_1 + w_2}, \quad i = 1, 2 \quad (3.3)$$

Apabila dibentuk lebih dari dua aturan, fungsi dapat diperluas dengan membagi w_i dengan jumlah total w untuk semua aturan.

4. Lapisan 4

Setiap simpul pada lapisan ini adalah simpul adaptif dengan fungsi simpul :

$$O_{4,i} = \bar{w}_i f_i = \bar{w}_i (p_i x + q_i y + r_i) \quad (3.4)$$

dengan \bar{w}_i adalah derajat perngaktifan ternormalisasi dari lapisan 3 dan parameter p, q, r menyatakan parameter konsekuen yang adaptif.

5. Lapisan 5

Fungsi lapisan ini adalah untuk menjumlahkan semua masukan. Fungsi simpul :

$$O_{5,i} = \sum_i \bar{w}_i f_i = \frac{\sum_i w_i f_i}{\sum_i w_i} \quad (3.5)$$

3.2 PROSES PEMBELAJARAN PADA ANFIS

Pada Gambar 3.1 simpul adaptif terdapat pada lapisan pertama dan keempat. Simpul pada lapisan pertama mengandung parameter premis yang nonlinier sedangkan pada lapisan keempat mengandung parameter konsekuen yang linier. Untuk memperbaharui parameter-parameter itu diperlukan metoda. Pembelajaran *hybrid* pada ANFIS, terdiri atas dua bagian yaitu arah maju dan arah mundur.

Pada arah maju, parameter premis dibuat tetap. Dengan menggunakan metode Galat Kuadrat Terkecil (RLSE), parameter konsekuen diperbaiki berdasarkan pasangan data masukan-keluaran. Metode RLSE dapat diterapkan karena parameter konsekuen yang diperbaiki adalah parameter linier. Metode RLSE akan mempercepat proses belajar hibrid [2]. Setelah parameter konsekuen didapat, masukan dilewatkan jaringan adaptif kembali dan hasil keluaran jaringan adaptif ini dibandingkan dengan keluaran yang sebenarnya. Pada arah mundur, parameter konsekuen dibuat tetap. Kesalahan yang terjadi antara keluaran jaringan adaptif dan keluaran sebenarnya dipropagasikan balik dengan menggunakan *gradient descent* untuk memperbaiki parameter premis. Tabel dibawah akan menerangkan proses pembelajaran hibrid ANFIS.

Tabel 3.1 Proses pembelajaran hibrid ANFIS

	Arah maju	Arah Mundur
Parameter premis	Tetap	Gradient descent
Parameter konsekuen	RLSE	Tetap
Sinyal	Keluaran Simpul	Laju Kesalahan

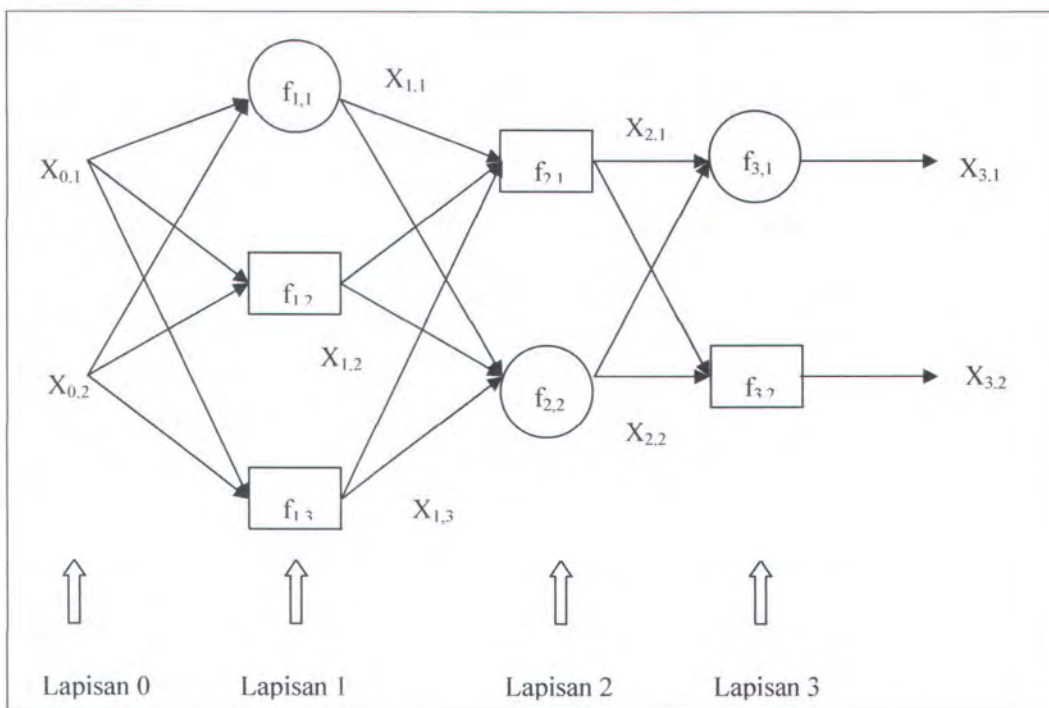
3.1.1 Proses Pembelajaran Backpropagation-error untuk Parameter

Premis

Parameter premis adalah parameter adaptif dengan proses pembelajaran yang menggunakan metode *gradient descent*. Misal terdapat sebuah sistem jaringan adaptif dinyatakan dengan L lapisan dan lapisan ke- l ($l = 0, 1, \dots, L$; $l = 0$ menyatakan lapisan masukan) mempunyai $N(l)$ simpul. Keluaran dan fungsi simpul ke- i [$i = 1, \dots, N(l)$] pada lapisan ke- l dinyatakan dengan $x_{l,i}$ dan $f_{l,i}$ seperti tampak pada gambar. Keluaran simpul merupakan fungsi dari sinyal yang masuk dan parameter sistem, maka diperoleh

$$x_{l,i} = f_{l,i}(x_{l-1,1}, \dots, x_{l-1,N(l-1)}, \alpha, \beta, \gamma, \dots) \quad (3.6)$$

dengan a, b, g adalah parameter simpul.

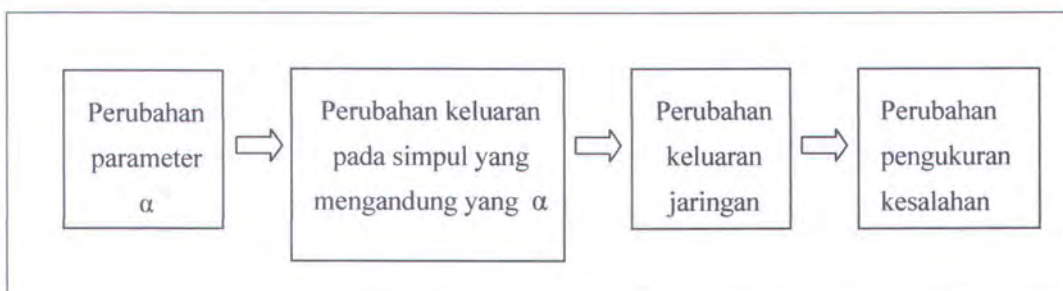


Gambar 3.2 Model jaringan syaraf feedforward

Misalkan sejumlah P pasangan data untuk proses belajar jaringan adaptif, selanjutnya dapat didefinisikan pengukuran kesalahan pada data latih ke- p adalah jumlah kuadrat kesalahan.

$$E_p = \sum_{k=1}^{N(L)} (d_k^p - x_{L,k}^p)^2 \quad (3.7)$$

dengan d_k^p adalah komponen ke- k dari vektor keluaran yang diinginkan dan $x_{L,k}^p$ adalah vektor keluaran aktual yang dihasilkan sistem jaringan adaptif dengan masukan dari vektor masukan ke- p dari P data belajar. Tujuan dari sistem adaptif adalah untuk meminimumkan pengukuran kesalahan pada persamaan diatas dengan mengubah parameter-parameter adaptif. Ketika E_p sama dengan nol, jaringan akan dapat menghasilkan keluaran yang sama dengan keluaran yang diinginkan dari P data pelatihan. Oleh karena itu untuk menghasilkan keluaran seperti yang diinginkan, keseluruhan pengukuran kesalahan (*error measure*) yang didefinisikan sebagai $E = \sum_{p=1}^P E_p$ harus di minimalkan. Dalam penggunaan turunan langkah (*steepest descent*) untuk meminimalkan pengukuran kesalahan, pertama harus didapat vektor kemiringan (*gradient vector*). Sebelum menghitung vektor kemiringan, perlu diketahui hubungan sebab dibawah ini:



Gambar 3.3 Diagram hubungan sebab akibat

panah \Rightarrow mengindikasikan hubungan sebab, dengan kata lain, perubahan kecil pada parameter α akan mempengaruhi keluaran simpul yang mengandung α . Dan hal ini juga akan mempengaruhi keluaran pada lapisan akhir begitu juga dengan pengukuran kesalahannya. Oleh karena itu, konsep dasar dalam menghitung vektor kemiringan adalah melewati informasi turunan dimulai dari lapisan keluaran dan berjalan mundur lapisan demi lapisan sampai ke lapisan masukan.

Dengan mendefinisikan sinyal kesalahan $\varepsilon_{l,i}$ sebagai turunan dari pengukuran kesalahan dan merupakan *ordered derivative* terhadap keluaran simpul ke- i , lapisan ke- l , maka *ordered derivative* dinotasikan dengan :

$$\varepsilon_{l,i} = \frac{\partial^+ E_p}{\partial x_{l,i}} \quad (3.8)$$

Sinyal kesalahan untuk simpul keluaran ke- i (pada lapisan L) dapat dihitung langsung dengan:

$$\varepsilon_{L,i} = \frac{\partial^+ E_p}{\partial x_{L,i}} = \frac{\partial E_p}{\partial x_{L,i}} \quad (3.9)$$

Jika pengukuran kesalahan seperti yang didefinisikan pada persamaan 3.8 maka persamaan 3.9 menjadi :

$$\varepsilon_{L,i} = -2(d_i^p - x_{L,i}^p) \quad (3.10)$$

Untuk simpul dalam pada lapisan l posisi ke- i , sinyal kesalahan dapat diperoleh menggunakan aturan rantai.

$$\varepsilon_{l,i} = \frac{\partial^+ E_p}{\partial x_{l,i}} = \sum_{m=1}^{N(l+1)} \frac{\partial^+ E_p}{\partial x_{l+1,m}} \frac{\partial f_{l+1,m}}{\partial x_{l,i}} = \sum_{m=1}^{N(l+1)} \varepsilon_{l+1,m} \frac{\partial f_{l+1,m}}{\partial x_{l,i}} \quad (3.11)$$

\uparrow

Sinyal kesalahan
pada lapisan l

\uparrow

Sinyal kesalahan
pada lapisan l



dengan $0 \leq l \leq L-1$. Sinyal kesalahan simpul dalam, pada lapisan ke- l dapat dinyatakan sebagai kombinasi linier dari sinyal kesalahan simpul pada lapisan ke $(l+1)$. Jadi untuk menghitung sinyal kesalahan pada simpul ke- i lapisan ke- l ($l < L$), pertama digunakan persamaan 3.11 untuk mendapatkan sinyal kesalahan pada lapisan keluaran kemudian persamaan diatas secara iteratif sampai mencapai lapisan yang diinginkan. Prosedur diatas disebut penjalaran balik (*backpropagation*) karena sinyal kesalahan dihitung secara mundur dari lapisan keluaran hingga lapisan masukan.

Vektor kemiringan didefinisikan sebagai *ordered derivative* dari pengukuran kesalahan terhadap tiap parameternya. Jika a adalah parameter simpul ke- l lapisan ke- l , maka diperoleh :

$$\frac{\partial^+ E_p}{\partial \alpha} = \frac{\partial E_p}{\partial x_{l,i}} \frac{\partial f_{l,i}}{\partial \alpha} = \varepsilon_{l,i} \frac{\partial f_{l,i}}{\partial \alpha} \quad (3.12)$$

Jika a merupakan parameter yang ada pada beberapa simpul maka persamaan diatas menjadi :

$$\frac{\partial^+ E_p}{\partial \alpha} = \sum_{x^* \in S} \frac{\partial^+ E_p}{\partial x^*} \frac{\partial f^*}{\partial \alpha} \quad (3.13)$$

dengan S merupakan himpunan simpul yang berisikan a sebagai parameter, sedangkan x^* dan f^* adalah keluaran dan fungsi dari simpul yang bersangkutan. Turunan masing-masing secara keseluruhan terhadap pengukuran kesalahan akan menghasilkan :

$$\frac{\partial^+ E_p}{\partial \alpha} = \sum_{p=1}^P \frac{\partial^+ E_p}{\partial \alpha} \quad (3.14)$$

Dengan metode gradient simple steepest descent, persamaan untuk memperbaiki

parameter α adalah :

$$\Delta\alpha = -\eta \frac{\partial^+ E}{\partial \alpha} \quad (3.15)$$

dengan η adalah laju proses belajar (learning rate) yang dinyatakan dengan:

$$\eta = \frac{k}{\sqrt{\sum_{\alpha} \left(\frac{\partial E}{\partial \alpha} \right)^2}} \quad (3.16)$$

dan k adalah ukuran langkah (*step size*) yang dapat diubah untuk mempercepat konvergensi. Jika k bernilai kecil, metode kemiringan (*gradient*) akan mendekati perkiraan garis kemiringan (*gradient path*) tetapi konvergensi akan lambat karena kemiringan harus dihitung setiap saat. Di lain pihak, jika k bernilai besar, konvergensi pada awalnya akan sangat cepat, tetapi algoritma akan berkisar optimal. Berdasarkan hal tersebut, untuk memperbaharui nilai k berdasarkan pada dua aturan *heuristic*, adalah:

1. Jika pengukuran kesalahan mengalami penurunan 4 kali secara berturut-turut, maka nilai k ditambah 10% dari nilainya.
2. Jika pengukuran kesalahan mengalami 2 kombinasi antara 1 naik dan 1 turun secara berturut-turut, maka nilai k dikurangi 10 dari nilainya.

Dalam tugas akhir ini fungsi keanggotaan yang digunakan adalah fungsi gauss2mf yaitu setengah bagian dari fungsi *Gaussian*. Rumus fungsi Gaussian adalah:

$$e^{-\frac{(x-\mu_i)^2}{2\sigma_i^2}} \quad (3.17)$$

Di dalam fungsi Gaussian ini terdapat dua parameter, yaitu center (μ) dan varian (σ). Untuk merubah parameter premis berdasarkan pada metode turunan gradient maka dicari turunan fungsi gaussian terhadap center dan variannya.. Berdasarkan

persamaan 3.12 dan 3.14, maka turunan fungsi Gaussian terhadap center adalah:

$$\frac{\partial E}{\partial \mu} = \varepsilon_{i,i} \exp\left\{-\frac{\|x - \mu_i\|^2}{2\sigma_i^2}\right\} \frac{(x - \mu_i)}{\sigma_i^2} \quad (3.18)$$

dan turunan fungsi Gaussian terhadap varian adalah:

$$\frac{\partial E}{\partial \sigma} = \varepsilon_{i,i} \exp\left\{-\frac{\|x - \mu_i\|^2}{2\sigma_i^2}\right\} \frac{\|x - \mu_i\|^2}{\sigma_i^3} \quad (3.19)$$

3.2.2 Proses Pembelajaran dengan RLSE untuk Parameter Konsekuen

Pada gambar struktur ANFIS jika nilai dari parameter premis tetap maka keluaran keseluruhannya dapat dinyatakan dengan kombinasi linier dari parameter konsekuen

$$\begin{aligned} f &= \bar{w}_1 f_1 + \bar{w}_2 f_2 \\ &= \bar{w}_1 (p_1 x + q_1 y + r_1) + \bar{w}_2 (p_2 x + q_2 y + r_2) \\ &= (\bar{w}_1 x)_1 p_1 + (\bar{w}_1 y)_1 q_1 + (\bar{w}_1)_1 r_1 + (\bar{w}_2 x)_1 p_2 + (\bar{w}_2 y)_1 q_2 + (\bar{w}_2)_1 r_2 \end{aligned} \quad (3.20)$$

Pada persamaan diatas terlihat parameter-parameter bagian konsekuen merupakan parameter linier terhadap keluaran sistem. Jika sejumlah N data belajar diterapkan pada persamaan 3.20, didapat :

$$\begin{aligned} (\bar{w}_1 x)_1 p_1 + (\bar{w}_1 y)_1 q_1 + (\bar{w}_1)_1 r_1 + (\bar{w}_2 x)_1 p_2 + (\bar{w}_2 y)_1 q_2 + (\bar{w}_2)_1 r_2 &= y_1 \\ \vdots & \\ (\bar{w}_1 x)_n p_1 + (\bar{w}_1 y)_n q_1 + (\bar{w}_1)_n r_1 + (\bar{w}_2 x)_n p_2 + (\bar{w}_2 y)_n q_2 + (\bar{w}_2)_n r_2 &= y_n \end{aligned} \quad (3.21)$$

persamaan 3.21 jika dinyatakan dengan persamaan matriks, akan berbentuk :

$$A \theta = y \quad (3.22)$$

dengan θ merupakan vektor yang elemen-elemennya merupakan himpunan parameter konsekuen yang berukuran $M \times 1$, A adalah matrik masukan dikali

dengan bobot yang berukuran $N * M$ dan y merupakan vektor keluaran yang elemen-elemennya $N * 1$ buah data keluaran sistem. Karena N (banyaknya data pada pembelajaran) biasanya lebih besar daripada M (banyaknya parameter linier) maka akan timbul masalah *overdeterminate* dan umumnya tidak ada pemecahan yang pasti untuk persamaan 3.22. Penyelesaian terbaik untuk θ adalah meminimumkan $\|A\theta - y\|^2$, dengan teori LSE (Least Square Estimator) berdasarkan persamaan 2.3, didapat θ^* :

$$\theta^* = (A^T A)^{-1} A^T y \quad (3.23)$$

dimana A^T adalah tranpose dari A , dan $(A^T A)^{-1} A^T$ adalah pseudo-invers dari A , jika $A^T A$ adalah *non singular*. Karena persamaan 3.23 adalah notasi yang singkat, maka akan dibutuhkan perhitungan yang rumit ketika berhubungan dengan matrik invers. Selain itu akan menimbulkan *ill-define* jika $A^T A$ adalah *singular*. Oleh karena itu diperlukan formula sekuensial yang menghitung LSE dari θ . Metode sekuensial dari LSE ini lebih efisien (khususnya jika M kecil). θ^* dihitung dengan rumus Recursive LSE (RLSE). Selanjutnya persamaan 3.23 menjadi :

$$\begin{cases} \theta_{i+1} = \theta_i + P_{i+1} a_{i+1} (y_{i+1}^T - a_{i+1}^T \theta_i) \\ P_{i+1} = P_i - \frac{P_i a_{i+1} a_{i+1}^T P_i}{1 + a_{i+1}^T P_i a_{i+1}}, \quad I = 0, 1, \dots, P-1 \end{cases} \quad (3.24)$$

dengan a_i^T adalah vektor baris dari matriks A , y_i adalah elemen ke- i dari y . P_i disebut matriks kovariansi yang didefinisikan dengan :

$$P_i = (A^T A)^{-1} \quad (3.25)$$

Dengan inialisasi awal $\theta_0 = 0$ dan $P_0 = \gamma I$, dimana γ adalah bilangan positif yang bernilai besar dan I adalah matrik identitas yang berdimensi $M * M$.

Abu Hurairah r.a berkata. Saya telah mendengar Rasulullah SAW. Bersabda:

*Dunia ini terkutuk dan segala isinya pun terkutuk kecuali dzikrullah (ta'at
pada Allah) dan yang serupa itu dan orang alim dan pelajar – Attarmidzi*

BAB IV

IMPLEMENTASI ANFIS

BAB IV

IMPLEMENTASI ANFIS

Dalam bab ini akan dibahas mengenai implementasi metode ANFIS ke dalam suatu aplikasi pengenalan wajah. Pembahasan ini meliputi lingkungan pembangunan aplikasi ANFIS, berupa masukan, keluaran, proses serta antarmuka aplikasi.

4.1 LINGKUNGAN PEMBANGUNAN APLIKASI ANFIS

Dalam bagian ini akan dijelaskan mengenai lingkungan pembangunan aplikasi ANFIS, meliputi perangkat lunak dan perangkat keras yang digunakan. Spesifikasi perangkat lunak dan perangkat keras yang digunakan dalam pembangunan aplikasi pengenalan wajah dengan menerapkan metode ANFIS ini dapat dilihat pada tabel berikut:

Tabel 4.1. Lingkungan Pembangunan Sistem Pengenalan Wajah

Perangkat Keras	Prosesor : Intel Pentim IV
	Memori : 512 MB
Perangkat Lunak	Sistem Operasi : Windows
	Perangkat Lunak Pembangun : Matlab 6.1

4.2 MASUKAN SISTEM PENGENALAN WAJAH

Masukan yang dibutuhkan oleh sistem ini dapat dilihat pada tabel 4.2.

Tabel 4.2. Keluaran yang dihasilkan oleh Proses Klasifikasi ANFIS

Jenis Masukan	Keterangan Masukan
Nama Basis Data	Nama Basis Data wajah yang akan dimasukkan kedalam sistem. Terdapat 3 buah Basis Data wajah baku yang digunakan, yaitu Basis Data Laboratorium Riset Ollivetti, Laboratorium Riset YALE dan The University of Bern (UB).
Metode pengambilan data	<p>Ada tiga pilihan metode pengambilan data, yaitu:</p> <ul style="list-style-type: none"> • <i>Leave- 10%-out</i> • <i>Leave- 25%-out</i> • <i>Leave- 50%-out</i> <p>Dimana <i>Leave- x %-out</i> berarti mengambil $(100-x)$ % data untuk proses pembelajaran (<i>training data</i>) sehingga menyisakan $x\%$ data yang akan digunakan untuk proses pengujian (<i>testing data</i>)</p>
Tujuan rata-rata kesalahan yang diinginkan (<i>error goal</i>)	Seberapa besar pengukuran kesalahan yang ingin dicapai
Jumlah iterasi (<i>epoch</i>)	Banyaknya <i>epoch</i> yang diinginkan (1 <i>epoch</i> merupakan gabungan antara metode pembelajaran 1 <i>forward pass</i> dan 1 <i>backward pass</i>).
Ukuran langkah (<i>step size</i>)	Ukuran langkah adalah jarak tiap transisi untuk mendekati arah kemiringan (<i>gradient</i>).

4.3 PROSES

Secara umum proses yang dilalui data yang masuk ke dalam sistem yang dibuat adalah mulai dari pembacaan data kemudian dilakukan sampai keluar hasilnya.

Proses-proses tersebut dapat dilihat pada gambar 4.1.



Gambar 4.1 Diagram Proses Pengklasifikasian dalam ANFIS

4.3.1 Proses Pembacaan Path file lokasi Gambar

Pada proses ini dilakukan pembacaan path file lokasi gambar yang disimpan dalam bentuk notepad. Penggalan program untuk bagian ini adalah sebagai berikut:

```
For i=1:jum_wajah
[gambar] = textread('file_gbr.txt','%s',i);
end
```

4.3.2 Proses Pembacaan Gambar dan diubah ke bentuk Matrik

Setelah path lokasi gambar diketahui, proses selanjutnya adalah mengambil gambar dan merubahnya ke bentuk matrik. Penggalan program untuk bagian ini adalah sebagai berikut:

```
For i=1:jum_wajah
B_test(:, :, i)=double(imread(gambar{i}));
end
```

4.3.3 Proses Utama

Proses Utama dari aplikasi ini ada tiga bagian, yaitu proses ekstraksi fitur global dengan metode PCA, proses ekstraksi fitur lokal dengan dekomposisi wavelet dan proses pembelajaran data dengan menggunakan ANFIS.

4.3.3.1 Proses ekstraksi fitur global dengan metode PCA

Pada proses ini terdiri dari empat bagian, yaitu proses mencari kovarian, mencari nilai eigen dan vektor eigen, mereduksi data dan transformasi *whitening*.

1. Proses mencari matrik kovarian dari matrik masukan

Proses ini adalah proses awal untuk menerapkan metode Analisa Komponen Utama, yaitu mencari matrik kovarian dari matrik masukannya. Sebelum mencari

matrik kovarian, terlebih dahulu dicari rata-rata tiap vektor. Dari rata-rata vektor tersebut kemudian dicari selisih antara matrik masukan dan rata-rata vektornya. Matrik kovarian dihasilkan dari perkalian antara matrik selisih dengan transpose dari matrik selisih itu sendiri. Penggalan program untuk bagian ini adalah sebagai berikut:

```
%mencari rata-rata vektor
ratal(baris,1)=0;
for j=1:baris;
    for i=1:baris_train;
        ratal(j,1)=ratal(j,1)+D_tran(j,i);
    end;
    mx(j,1)=ratal(j,1)/baris_train;
end;
%mencari selisih vektor
for j=1:baris;
    for i=1:baris_train;
        selisih_vek(j,i)=D_tran(j,i)-mx(j,1);
    end;
end;
%Mencari matrik kovarian
n=baris_train-2;
Scatter=selisih_vek.'*selisih_vek;
```

2. Proses mencari nilai eigen dan vektor eigen

Dari matrik kovarian yang dihasilkan, kemudian dicari nilai eigen dan vektor eigennya. Penggalan program untuk bagian ini adalah sebagai berikut:

```
[U,S,V]=svd(Scatter);
S(n:baris_train,:)=[];
S(:,n:baris_train)=[];
```



3. Proses mereduksi Data

Mereduksi data dilakukan berdasarkan pada nilai eigennya. Nilai eigen diurutkan berdasarkan nilai yang paling besar. Ada beberapa cara untuk memotong atau mereduksi matrik, salah satunya adalah memilih nilai eigen yang lebih besar dari

1. Penggalan program untuk bagian ini adalah sebagai berikut:

```
Q=D_tran*U;
%potong Q sesuai nilai eigen
Q(:,n:baris_train)=[];
PCA_training=Q.'*D_tran;
```

4. Proses Transformasi Whitening

Untuk hasil yang lebih baik, dilakukan transformasi whitening terhadap hasil dari Analisa Komponen Utama. Transformasi Whitening dilakukan dengan mengalikan matrik hasil reduksi dengan invers dari akar nilai eigen. Penggalan program untuk bagian ini adalah sebagai berikut:

```
% mencari invers dari akar nilai eigen
S_inv = inv(sqrt(S));
% dikalikan dengan hasil matrik reduksi
PCA_training_akhir = S_inv*PCA_training;
```

4.3.3.2 Proses ekstraksi fitur lokal dengan dekomposisi wavelet

Proses ini merupakan aplikasi dari metode Analisa Paket Wavelet 2-D, yang terdiri dari dua bagian, yaitu:

1. Proses dekomposisi Matrik masukan

Pada proses ini dilakukan dekomposisi matrik masukan dengan menggunakan filter haar. Koefisien filter haar adalah: 0.70710678 untuk filter *low-pass* dan -

0.70710678 untuk filter *high-pass*. Penggalan program untuk dekomposisi matrik adalah sebagai berikut:

```
[jum_baris,jum_kolom]=size(data);
k=uint8(jum_baris/2);
l=double(k);
n=uint8(jum_baris/4);
m=double(k);

i=1;
filter_h=0.70710678;
filter_g=-0.70710678;
while i<jum_baris-1
    filter(i,i)=filter_h;
    filter(i,i+1)=filter_h;
    filter(i+1,i)=filter_h;
    filter(i+1,i+1)=filter_g;
    i=i+2;
end
if mod(jum_baris,2)==0
    filter(i,i)=filter_h;
    filter(i,i+1)=filter_h;
    filter(i+1,i)=filter_h;
    filter(i+1,i+1)=filter_g;
else
    filter(i,i)=filter_h;
end
wave=filter*data;
%brs ganjil unt low pass
%dan brs genap unt high pass
[baris,kolom]=size(wave);
low=wave(1:2:baris,:);
high=wave(2:2:baris,:);
%baris genap low pass dan high pass dihilangkan
j=2;
```

```

k=uint8(jum_kolom/2);
n=double(k);
while j<n+2;
low(:,j)=[];
high(:,j)=[];
j=j+1;
end

```

2. Proses mencari energi atau bobot

Hasil dari dekomposisi sebuah gambar wajah adalah matrik sebanyak 16 buah.

Untuk setiap matrik dicari energi distribusi atau bobotnya. Penggalan program untuk bagian ini adalah sebagai berikut:

```

for i=1:jum_wajah
    w(i,1,1)=g(i,1,1)*g(i,2,1);
    w(i,2,1)=g(i,1,2)*g(i,2,1);
    w(i,3,1)=g(i,1,1)*g(i,2,2);
    w(i,4,1)=g(i,1,2)*g(i,2,2);
end
jum_w=sum(w.').' ;
w_norm(jum_wajah,4)=0;
for i=1:jum_wajah;
    for j=1:4;
        w_norm(i,j)=w(i,j)/jum_w(i,1);
    end
end
end

```

4.3.3.3 Proses Pembelajaran Data

Pada proses pembelajaran ini, terdapat dua bagian, yaitu pembelajaran arah mundur (*backward pass*) dan arah maju (*forward pass*). Untuk arah mundur menggunakan metode Gradient descent dan arah maju menggunakan metode

RLSE. Sebelum dilakukan pembelajaran data terlebih dahulu dilakukan inialisasi awal nilai keanggotaan, bobot dan proses defuzzifikasinya.

1. Proses inialisasi nilai keanggotaan

Pada proses ini dilakukan pencarian nilai keanggotaan awal yang kemudian digunakan sebagai masukan bagi aturan fuzzy. Untuk mencari nilai keanggotaan, terlebih dahulu di cari center dan varian dari fungsi Gaussian. Penggalan program untuk mencari nilai keanggotaannya adalah sebagai berikut:

```
for j=1:n
    for i=1:jum_wajah
        if gabung_normal(i,j)< premis(j,1)
            g(i,j,1)=gaussmf(gabung_normal(i,j), [premis(j,3)
                premis(j,1)]);
        else
            g(i,j,1)=1;
        end
        if gabung_normal(i,j)< premis(j,2)
            g(i,j,2)=gaussmf(gabung_normal(i,j), [premis(j,4)
                premis(j,2)]);
        else
            g(i,j,2)=1;
        end
    end
end
```

2. Proses mencari bobot dan normalisasi

Pada proses ini dilakukan perkalian nilai keanggotaan berdasarkan aturan fuzzy, yaitu kombinasi dari nilai keanggotaan. Penggalan program untuk mencari nilai keanggotaannya adalah sebagai berikut:

```

function [w_norm]=bobot(g,jum_wajah);
% Dikali (representasi AND) sesuai RULE
for i=1:jum_wajah
    w(i,1,1)=g(i,1,1)*g(i,2,1);
    w(i,2,1)=g(i,1,2)*g(i,2,1);
    w(i,3,1)=g(i,1,1)*g(i,2,2);
    w(i,4,1)=g(i,1,2)*g(i,2,2);
end
% jumlah w tiap wajah
jum_w=sum(w.').';
%w dinormalisasi
w_norm(jum_wajah,4)=0;
for i=1:jum_wajah;
    for j=1:4;
        w_norm(i,j)=w(i,j)/jum_w(i,1);
    end
end
end

```

3. Proses defuzzifikasi

Proses defuzzifikasi ini, dilakukan untuk mencari nilai keluaran tunggal, dalam hal ini kelas dari data masukan. Penggalan program untuk bagian ini adalah sebagai berikut:

```

for i=1:jum_wajah
    for j=1:4
        for k=1:n
            coba(j,k,i)=gabung_normal(i,k)*w_norm(i,j);
        end
    end
end
for i=1:jum_wajah
    coba(:,3,i)=w_norm(i,:);
end

```



```

hasil(jum_wajah,12)=0;
for i=1:jum_wajah;
    hasil(i,:,1)=[coba(1,:,i) coba(2,:,i) coba(3,:,i)
coba(4,:,i)];
end

```

4. Pembelajaran arah mundur

Pada proses ini dilakukan pencarian dan pengubahan parameter premis dengan menggunakan metode *Gradien Descent*. Fungsi keanggotaan yang digunakan adalah fungsi Gaussian, sehingga ada dua parameter premis yang perlu diubah untuk tiap iterasi, yaitu center dan varian. Penggalan program untuk mencari center dan varian baru adalah sebagai berikut:

```

% mencari center baru
sinyal_eror5=-2*sum(temporal2);
k=0.1;
for j=1:n
    for i=1:jum_wajah
        if gabung_normal(i,j)< premis(j,1,tambah)
gbaru(i,j,1)=turunan_gaus_center(gabung_normal(i,j),[premis(
j,3,tambah) premis(j,1,tambah)]);
        else
            gbaru(i,j,1)=1;
        end
        if gabung_normal(i,j)< premis(j,2,tambah)
gbaru(i,j,2)=turunan_gaus_center(gabung_normal(i,j),[premis(
j,4,tambah) premis(j,2,tambah)]);
        else
            gbaru(i,j,2)=1;
        end
    end
end
end

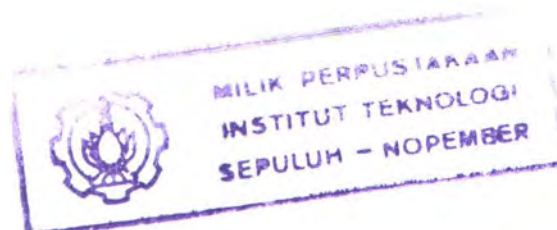
```

```

w_ratal=w_norm(:, :, tambah);
w_rata=sum(w_ratal. ' ');
for i=1:jum_wajah
    lamda_c(i, :, 1)=sinyal_eror5(1,1)*gbaru(i, :, 1);
    lamda_c(i, :, 2)=sinyal_eror5(1,1)*gbaru(i, :, 2);
end
lamda_center1=sum(lamda_c(:, :, 1));
lamda_center2=sum(lamda_c(:, :, 2));
for i=1:n
    l_c1(1,i)=lamda_center1(1,i)*lamda_center1(1,i);
    l_c2(1,i)=lamda_center2(1,i)*lamda_center2(1,i);
end
jml_lc1=sqrt(sum(l_c1));
jml_lc2=sqrt(sum(l_c2));
lamda_center1=(k/jml_lc1)*lamda_center1;
lamda_center2=(k/jml_lc2)*lamda_center2;
premis_baru(:, 1)=premis(:, 1, tambah)-lamda_center1. ' ';
premis_baru(:, 2)=premis(:, 2, tambah)-lamda_center2. ' ';

% mencari varian baru
ksig=0.1;
for j=1:n
    for i=1:jum_wajah
        if gabung_normal(i,j) < premis(j,1,tambah)
            g_sigma(i,j,1)=turunan_gaus_sigma(gabung_normal(i,j), [premis
(j,3,tambah) premis(j,1,tambah)]);
        else
            g_sigma(i,j,1)=1;
        end
        if gabung_normal(i,j) < premis(j,2,tambah)
            g_sigma(i,j,2)=turunan_gaus_sigma(gabung_normal(i,j), [premis
(j,4,tambah) premis(j,2,tambah)]);
        else
            g_sigma(i,j,2)=1;
        end
    end
end

```



```

        end
    end
    for i=1:jum_wajah
        lamda_s(i,:,1)=sinyal_eror5(1,1)*g_sigma(i,:,1);
        lamda_s(i,:,2)=sinyal_eror5(1,1)*g_sigma(i,:,2);
    end
    lamda_s1=sum(lamda_s(:,:,1));
    lamda_s2=sum(lamda_s(:,:,2));
    for i=1:n
        l_s1(1,i)=lamda_s(1,i)*lamda_s1(1,i);
        l_s2(1,i)=lamda_s(1,i)*lamda_s2(1,i);
    end
    jml_ls1=sqrt(sum(l_s1));
    jml_ls2=sqrt(sum(l_s2));
    lamda_s1=(ksig/jml_ls1)*lamda_s1;
    lamda_s2=(ksig/jml_ls2)*lamda_s2;
    premis_baru(:,3)=premis(:,3,tambah)-lamda_s1.';
    premis_baru(:,4)=premis(:,4,tambah)-lamda_s2.';
    premis(:, :, tambah1)=premis_baru;

```

5. Pembelajaran arah maju

Pada proses ini dilakukan pencarian parameter konsekuen dengan menggunakan metode RLSE. Penggalan program untuk bagian ini adalah sebagai berikut:

```

% X adalah parameter konsekuen yang baru
X1(80,1)=0;
S1(:, :)=100000*eye(80);
for i=1:280
    S2(:, :)=S1(:, :)-
    ((S1(:, :)*hasil_baru(i,:).'*hasil_baru(i,:)*S1(:, :))/(1+hasil_baru(i,:)*S1(:, :)*hasil_baru(i,:).'));
    selisih=0;
    kali=hasil_baru(i,:)*X1(:,i);

```



```

selisih(1,1)=output_training(i,1)-kali(1,1);
tempnya(:,1)=S2(:,1)*hasil_baru(i,:).'*selisih
(1,1);
X1(:,i+1)=X1(:,i)+tempnya(:,1);
S1=S2;
end
X=X1;

```

4.4 KELUARAN SISTEM PENGENALAN WAJAH

Keluaran yang dihasilkan oleh proses pengklasifikasian ANFIS dapat dilihat pada tabel berikut:

Tabel 4.3. Keluaran yang dihasilkan oleh Proses Klasifikasi ANFIS

Jenis Keluaran	Keterangan
Akurasi Pengklasifikasian Data	Akurasi ini adalah prosentase ketepatan pengklasifikasian data yang diperoleh dari pengujian. Ada 2 macam akurasi yang dihasilkan, yaitu akurasi pengklasifikasian data pembelajaran dan akurasi pengklasifikasian data uji ke dalam kelas-kelas.
Rata-rata kesalahan	Rata-rata kesalahan ini adalah <i>mean square error</i> (RMSE) yang dihasilkan dari pengurangan antara keluaran yang diinginkan terhadap keluaran yang dihasilkan sistem. Rata-rata kesalahan ini, bertujuan untuk mencari parameter premis baru, yang akan memperkecil rata-rata kesalahan pada iterasi (<i>epoch</i>) selanjutnya. Ada 2 macam rata-rata kesalahan yang dihasilkan, yaitu rata-rata

	kesalahan data pembelajaran dan rata-rata kesalahan data uji.
Grafik	<p>Dalam sistem ini menghasilkan empat grafik, yaitu:</p> <ol style="list-style-type: none"> 1. Grafik rata-rata kesalahan (RMSE) untuk data pembelajaran dan data uji 2. Grafik Akurasi Data 3. Grafik keluaran sistem untuk data pembelajaran 4. Grafik keluaran sistem untuk data uji 5. Grafik derajat keanggotaan untuk data pembelajaran 6. Grafik derajat keanggotaan untuk data uji.

4.1 ANTARMUKA

Ada 3 form yang digunakan dalam aplikasi ini, yaitu form untuk memasukkan masukan, form untuk menampilkan keluaran dan form untuk menampilkan grafik.

Gambar 4.2. Form Masukan Data

4.1.1 Form Masukan

Form ini dibagi menjadi dua bagian, yaitu bagian Data dan bagian masukan untuk Sistem Inferensi Fuzzy (FIS). Bagian Data terdiri dari dua bagian juga, yaitu:

1. Bagian pemilihan Basis Data

Bagian ini terdiri dari sebuah combobox, yang terdiri dari Basis Data wajah yang dapat dipilih untuk diklasifikasikan.

2. Bagian Pemilihan Metode Pengambilan Data

Ada tiga pilihan metode pengambilan data, yaitu *leave-10%-out*, *leave-25%-out* dan *leave-50%-out*.

3. Tombol Load Data

Jika tombol ini ditekan, data akan dibaca kemudian dimasukkan ke dalam variabel dan dihitung berapa besar data pembelajaran dan data ujinya. Selain itu data juga mengalami proses reduksi dan dekomposisi.



Gambar 4.3. Bagian pengambilan data pada Form Masukan Data

Bagian Masukan untuk Sistem Inferensi Fuzzy ada 3 bagian, yaitu:

1. Rata-rata Kesalahan yang ingin dicapai

Bagian ini terdiri dari textbox yang digunakan untuk memasukkan nilai rata-rata kesalahan yang ingin dicapai.

2. Iterasi yang akan dilakukan

Bagian ini terdiri dari textbox yang digunakan untuk memasukkan jumlah iterasi

yang akan dilakukan.

3. Ukuran langkah (*step size*)

Bagian ini terdiri dari textbox yang digunakan untuk memasukkan nilai ukuran langkah (*step size*), nilai yang dapat menghasilkan keluaran yang paling bagus berdasarkan percobaan adalah 0.1

The screenshot shows a form titled 'Fuzzy FIS'. It contains three input fields with labels and values:

Label	Value
Kecepatan gerak robot dikayuh	0
Jarak :	500
Ukuran langkah :	0.1

Gambar 4.4. Bagian masukan data untuk FIS pada Form Masukan Data

4.1.2 Form Keluaran

Pada form ini ditampilkan hasil percobaan yang telah dilakukan. Hasil yang ditampilkan antara lain: Akurasi Pengklasifikasian data, rata-rata kesalahan data pembelajaran dan data uji serta grafik-grafik. Gambar form dapat dilihat pada gambar 4.5 dibawah ini.

The screenshot shows a window titled 'Form Hasil Klasifikasi'. It contains a text area with the following text:

```

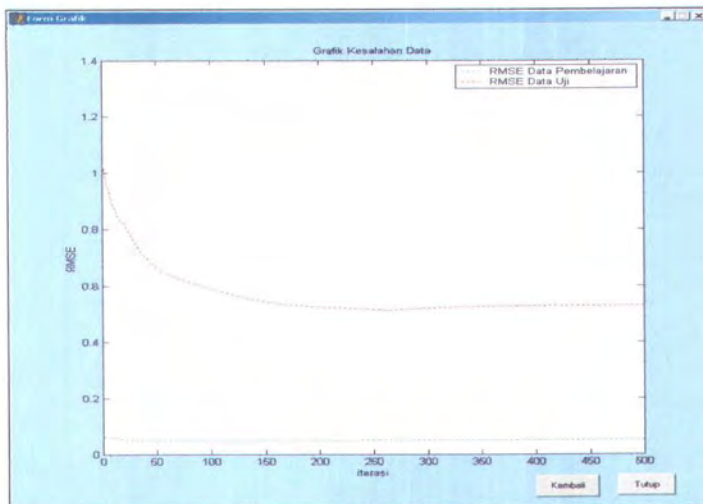
Hasil Program Pengklasifikasi:
>> RMSE Pembelajaran = 0.059331
>> RMSE Uji = 0.44813
  
```

Below the text area is a table titled 'Tabel Keluaran' with the following data:

Label	Value	Unit
Akurasi Pembelajaran :	99.0596	%
Akurasi Uji :	90	%
Skala :	Tingkat Kesalahan Data - RMSE	

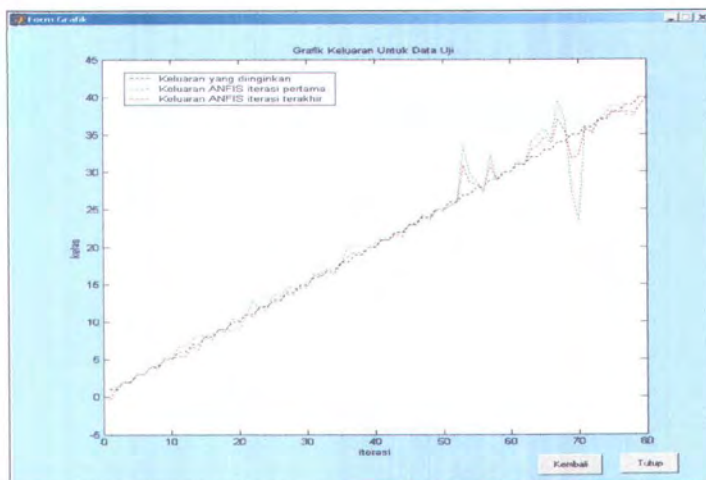
Gambar 4.5. Form Menampilkan Keluaran

Gambar form grafik rata-rata kesalahan data dapat dilihat pada gambar 4.6 dibawah ini:



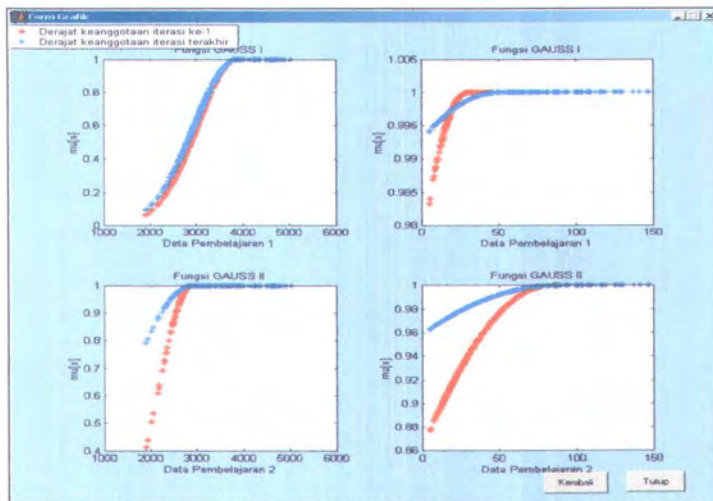
Gambar 4.6. Grafik Kesalahan Data- RMSE

Grafik untuk melihat hasil keluaran sistem terhadap keluaran yang diinginkan, dapat dilihat seperti pada gambar 4.7 dibawah ini:



Gambar 4.7. Grafik Keluaran Sistem

Untuk melihat perubahan atau pergeseran derajat keanggotaan data, terdapat grafik derajat keanggotaan yang dapat dilihat seperti pada gambar 4.8 dibawah ini:



Gambar 4.8. Grafik Derajat Keanggotaan

Allah, tiada Tuhan (yang berhak disembahi) melainkan Dia Yang Maha hidup, kekal lagi terus menerus mengurus (makhluk-Nya), tidak mengantuk dan tidak tidur. Kepunyaan-Nya apa yang dilangit dan dibumi. Tiada yang dapat memberi manfaat di sisi Allah tanpa izin-Nya. Allah Mengetahui apa-apa yang di hadapan mereka dan di belakang mereka, dan mereka tidak mengetahui apa-apa dari ilmu Allah melainkan apa yang dikehendaki-Nya. Kursi Allah meliputi langit dan bumi. Dan Allah tidak merasa berat memelihara keduanya, dan Allah Mahatinggi lagi Mahabesar -- (Al-Baqarah 255)

BAB V

UJI COBA DAN EVALUASI

BAB V

UJI COBA DAN EVALUASI

Pada bab ini akan dijelaskan mengenai uji coba dan evaluasi yang telah dilakukan. Pembahasan yang dikemukakan meliputi lingkungan uji coba, uji coba dengan data Ollivetti, YALE dan BERN.

5.1 LINGKUNGAN UJI COBA

Pada bagian ini akan dijelaskan mengenai lingkungan ujicoba aplikasi ANFIS yang meliputi perangkat keras dan perangkat lunak yang digunakan. Spesifikasi perangkat keras dan perangkat lunak yang digunakan dalam pengujian ini dapat dilihat pada tabel berikut ini:

Tabel 5.1. Lingkungan Pengujian Sistem Pengenalan Wajah

Perangkat Keras	Prosesor : Intel Pentim IV
	Memori : 512 MB
Perangkat Lunak	Sistem Operasi : Windows
	Perangkat Lunak Pembangun : Matlab 6.1

5.2 PELAKSANAAN UJI COBA DAN EVALUASI

Pengujian aplikasi ANFIS yang dikembangkan pada penelitian ini dilakukan untuk menguji kebenaran serta performansi program. Uji kebenaran dilakukan dengan menggunakan data uji berukuran kecil untuk mengetahui apakah perangkat lunak dapat memberikan hasil yang valid. Sedangkan uji kinerja

Tabel 5.3 Hasil pengujian Basis Data BERN dengan Metode ANFIS

Nomor Subyek	Leave-50%-out		Leave-75%-out		Leave-10%-out	
	Data Uji	Jumlah Data Yang Dikenali Benar	Data Uji	Jumlah Data Yang Dikenali Benar	Data Uji	Jumlah Data Yang Dikenali Benar
1	5	5	3	3	1	1
2	5	5	3	3	1	1
JUMLAH PROSENTASE		10		6		2
		100%		100%		100%

5.2.2 Uji Kinerja

Untuk menguji kinerja perangkat lunak ini, maka digunakan data berukuran besar yang telah memiliki nilai optimal, kemudian dicari tingkat Akurasinya.

5.2.2.1 Uji coba dan Evaluasi terhadap Basis Data OLLIVETTI

Ujicoba dan evaluasi terhadap sistem ini dilakukan dengan menggunakan citra wajah yang berukuran 92 x 112 piksel. Citra wajah tersebut berada dalam bentuk grey scale, dengan tingkat keabuan 256. Data uji coba diambil dari Basis Data wajah yang dimiliki oleh laboratorium riset Ollivetti. Basis Data ini terdiri dari 400 obyek wajah dari 40 orang dengan 10 variasi untuk setiap orangnya. Data ini diindeks dengan angka 1 sampai dengan 10. Sebagian dari data ini digunakan untuk proses pembelajaran dan sisanya sebagai data uji coba.

5.2.1 Hasil Ujicoba Data Whitening

Pengujian pada sistem pengenalan wajah yang dikembangkan pada penelitian ini dilakukan dengan memisahkan data citra wajah pada suatu Basis Data menjadi

dua himpunan yang saling pisah (*disjoint*) yaitu himpunan citra pelatihan dan citra pengujian. Penghitungan prosentasi keberhasilan pengenalan dilakukan pada himpunan citra pengujian. Berikut adalah tabel pembelajaran pada Basis Data OLLIVETTI.

Tabel 5.4 Hasil pembelajaran Basis Data OLLIVETTI dengan Metode ANFIS

Nomor Subyek	Leave-50%-out		Leave-75%-out		Leave-10%-out	
	Jumlah Data Pembelajaran	Jumlah Data Yang Dikenali Benar	Jumlah Data Pembelajaran	Jumlah Data Yang Dikenali Benar	Jumlah Data Pembelajaran	Jumlah Data Yang Dikenali Benar
1	5	5	7	7	9	9
2	5	5	7	7	9	9
3	5	5	7	7	9	9
4	5	5	7	7	9	9
5	5	5	7	7	9	9
6	5	5	7	7	9	9
7	5	5	7	7	9	9
8	5	5	7	7	9	9
9	5	5	7	7	9	9
10	5	5	7	6	9	9
11	5	5	7	7	9	9
12	5	5	7	7	9	9
13	5	5	7	7	9	9
14	5	5	7	7	9	9
15	5	5	7	7	9	9
16	5	5	7	7	9	9
17	5	5	7	7	9	9
18	5	5	7	7	9	9
19	5	5	7	7	9	9
20	5	5	7	7	9	9
21	5	5	7	7	9	9

22	5	5	7	7	9	9
23	5	5	7	7	9	9
24	5	5	7	7	9	9
25	5	5	7	7	9	9
26	5	5	7	7	9	9
27	5	5	7	7	9	9
28	5	5	7	7	9	9
29	5	5	7	7	9	8
30	5	5	7	7	9	9
31	5	5	7	7	9	9
32	5	5	7	7	9	9
33	5	5	7	7	9	7
34	5	5	7	7	9	9
35	5	5	7	7	9	9
36	5	5	7	7	9	9
37	5	5	7	7	9	9
38	5	5	7	7	9	9
39	5	5	7	6	9	5
40	5	5	7	7	9	9
JUMLAH		200		278		353
PROSENTASE		100%		99.2%		98%

Berikut ini akan diberikan hasil yang diperoleh dari percobaan yang dilakukan.

Hasil dari pelatihan terhadap data pembelajaran dimasukkan kedalam data uji, kemudian hasilnya dihitung.

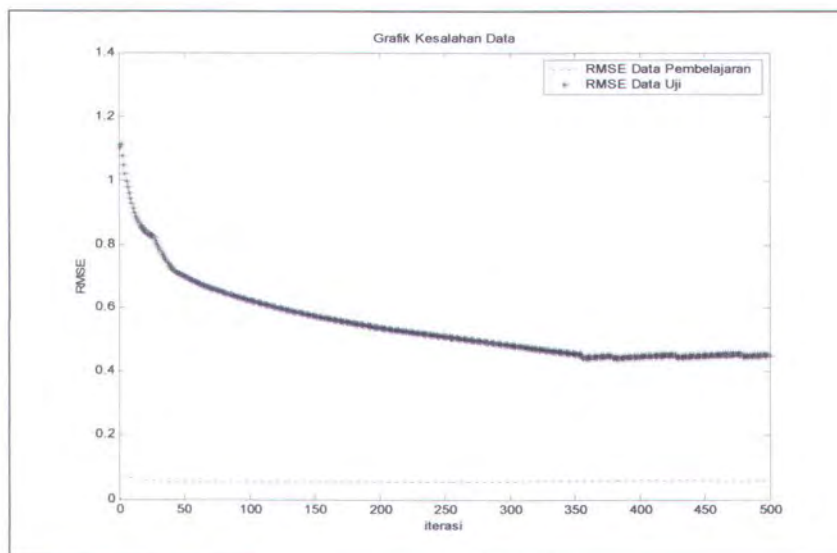
Tabel 5.5 Hasil pengujian Basis Data OLLIVETTI dengan Metode ANFIS

Nomor Subyek	Leave-50%-out		Leave-75%-out		Leave-10%-out	
	Data Uji	Jumlah Data Yang Dikenali Benar	Data Uji	Jumlah Data Yang Dikenali Benar	Data Uji	Jumlah Data Yang Dikenali Benar
1	5	1	3	1	1	1
2	5	5	3	2	1	1
3	5	5	3	3	1	1

4	5	5	3	3	1	1
5	5	5	3	3	1	1
6	5	4	3	3	1	0
7	5	5	3	1	1	0
8	5	5	3	3	1	1
9	5	5	3	3	1	1
10	5	5	3	3	1	1
11	5	5	3	3	1	1
12	5	5	3	3	1	1
13	5	0	3	3	1	1
14	5	1	3	3	1	1
15	5	5	3	3	1	1
16	5	5	3	3	1	1
17	5	4	3	3	1	1
18	5	5	3	2	1	0
19	5	5	3	1	1	1
20	5	5	3	3	1	1
21	5	5	3	3	1	1
22	5	5	3	3	1	1
23	5	5	3	3	1	1
24	5	5	3	3	1	1
25	5	5	3	3	1	1
26	5	5	3	2	1	1
27	5	0	3	0	1	0
28	5	4	3	3	1	1
29	5	2	3	1	1	1
30	5	5	3	3	1	1
31	5	1	3	3	1	1
32	5	2	3	0	1	0
33	5	0	3	0	1	0
34	5	1	3	0	1	1
35	5	0	3	0	1	0
36	5	2	3	2	1	1
37	5	4	3	2	1	1

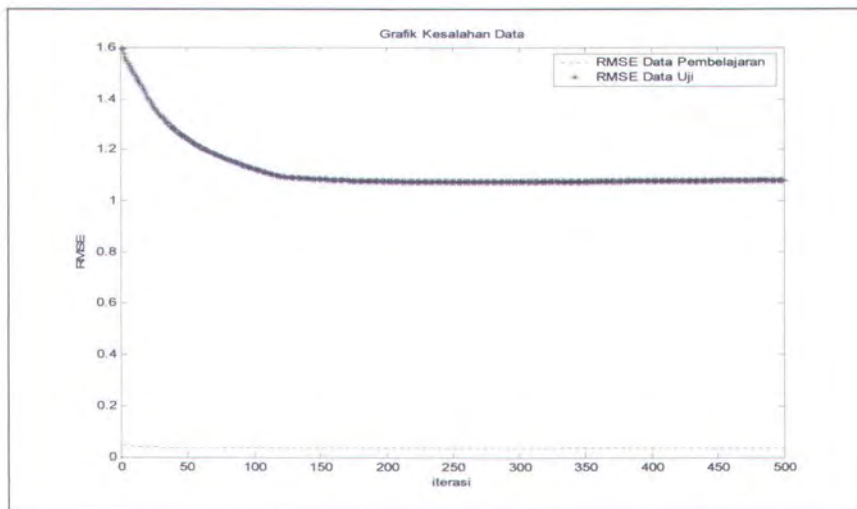
38	5	5	3	0	1	1
39	5	1	3	0	1	0
40	5	3	3	0	1	1
JUMLAH		145		86		32
PROSENTASE		72.5%		71.66%		80%

Dari pembelajaran dan pengujian tersebut didapatkan rata-rata kesalahan. Grafik RMSE untuk data OLLIVETTI dengan metode pengambilan data *Leave-10%-out* dapat dilihat pada gambar 5.1

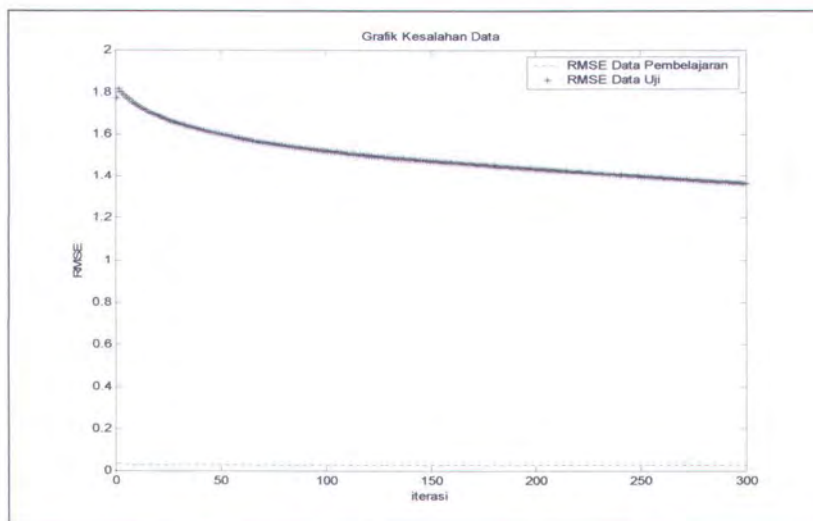


Gambar 5.1. Grafik RMSE untuk Data Ollivetti Leave-10%-out

Sedangkan untuk data OLLIVETTI dengan metode pengambilan *Leave-25%-out* dan *Leave-50%-out* grafik RMSE-nya dapat dilihat pada gambar 5.2 dan gambar 5.3



Gambar 5.2. Grafik RMSE untuk Data Ollivetti Leave-25%-out



Gambar 5.3. Grafik RMSE untuk Data Ollivetti Leave-50%-out

5.2.2 Hasil Ujicoba Data tanpa whitening

Berikut ini akan diberikan hasil yang diperoleh dari percobaan untu data yang tidak di-*whitening*. Pengujian dilakukan terhadap data pembelajaran dan data uji, kemudian hasilnya dihitung.

Tabel 5.6 Hasil pembelajaran Basis Data OLLIVETTI (tanpa whitening) dengan Metode ANFIS

Nomor Subyek	Leave-50%-out		Leave-75%-out		Leave-10%-out	
	Jumlah Data Pembelajaran n	Jumlah Data Yang Dikenali Benar	Jumlah Data Pembelajaran	Jumlah Data Yang Dikenali Benar	Jumlah Data Pembelajaran n	Jumlah Data Yang Dikenali Benar
1	5	5	7	7	9	9
2	5	5	7	7	9	9
3	5	3	7	5	9	7
4	5	3	7	6	9	6
5	5	5	7	7	9	9
6	5	3	7	4	9	7
7	5	4	7	5	9	5
8	5	4	7	6	9	8
9	5	1	7	0	9	2
10	5	4	7	5	9	6
11	5	3	7	6	9	7
12	5	1	7	2	9	3
13	5	4	7	6	9	7
14	5	3	7	3	9	5
15	5	0	7	0	9	0
16	5	2	7	3	9	3
17	5	0	7	0	9	0
18	5	4	7	6	9	8
19	5	3	7	1	9	1
20	5	2	7	2	9	3
21	5	2	7	2	9	2
22	5	3	7	4	9	4
23	5	1	7	0	9	1
24	5	1	7	1	9	1
25	5	0	7	0	9	0
26	5	1	7	1	9	1
27	5	1	7	1	9	0

28	5	0	7	0	9	0
29	5	3	7	3	9	3
30	5	2	7	3	9	3
31	5	0	7	0	9	0
32	5	2	7	3	9	3
33	5	2	7	5	9	5
34	5	1	7	2	9	2
35	5	0	7	1	9	3
36	5	0	7	0	9	0
37	5	1	7	0	9	0
38	5	0	7	0	9	0
39	5	1	7	1	9	2
40	5	0	7	0	9	0
JUMLAH		126		194		257
PROSENTASE		63%		69.28%		71.38%

Berikut ini akan diberikan hasil yang diperoleh dari percobaan yang dilakukan.

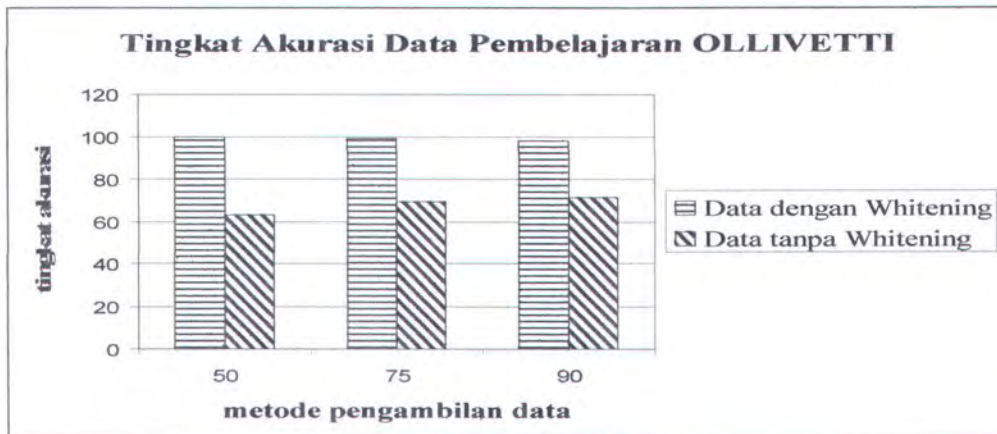
Hasil dari pelatihan terhadap data pembelajaran dimasukkan kedalam data uji, kemudian hasilnya dihitung.

Tabel 5.7 Hasil pengujian Basis Data OLLIVETTI (tanpa whitening) dengan Metode ANFIS

Nomor Subyek	Leave-50%-out		Leave-75%-out		Leave-10%-out	
	Data Uji	Jumlah Data Yang Dikenali Benar	Data Uji	Jumlah Data Yang Dikenali Benar	Data Uji	Jumlah Data Yang Dikenali Benar
1	5	5	3	3	1	1
2	5	5	3	3	1	1
3	5	0	3	0	1	1
4	5	2	3	2	1	1
5	5	1	3	0	1	0
6	5	0	3	0	1	0
7	5	3	3	3	1	0
8	5	0	3	0	1	1

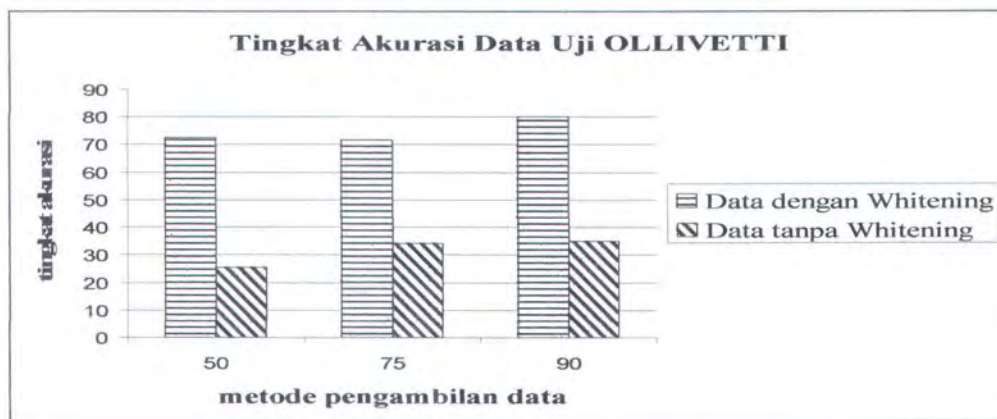
9	5	1	3	0	1	1
10	5	1	3	2	1	0
11	5	2	3	1	1	0
12	5	3	3	2	1	1
13	5	2	3	2	1	1
14	5	0	3	0	1	1
15	5	4	3	0	1	0
16	5	0	3	0	1	0
17	5	0	3	2	1	0
18	5	4	3	1	1	0
19	5	3	3	0	1	0
20	5	0	3	0	1	0
21	5	1	3	3	1	1
22	5	3	3	0	1	0
23	5	0	3	2	1	0
24	5	0	3	0	1	0
25	5	1	3	2	1	0
26	5	2	3	3	1	1
27	5	0	3	0	1	0
28	5	0	3	1	1	0
29	5	0	3	1	1	0
30	5	0	3	1	1	1
31	5	1	3	3	1	0
32	5	0	3	0	1	0
33	5	0	3	0	1	0
34	5	0	3	0	1	0
35	5	0	3	0	1	0
36	5	4	3	1	1	0
37	5	0	3	0	1	0
38	5	0	3	0	1	1
39	5	1	3	1	1	0
40	5	2	3	2	1	1
JUMLAH		145		41		14
PROSENTASE		72.5%		34.16%		35%

Perbandingan hasil ujicoba antara data yang di-*whitening* dengan data yang tidak di-*whitening* pada Basis Data OLLIVETTI dapat dilihat pada gambar 5.4 dan 5.5 dibawah ini:



Gambar 5.4. Grafik Akurasi Data Pembelajaran OLLIVETTI

Untuk tingkat akurasi data ujinya dapat dilihat pada gambar dibawah ini:



Gambar 5.5. Grafik Akurasi Data Uji OLLIVETTI

5.2.2.2 Uji coba dan Evaluasi terhadap Basis Data YALE

Ujicoba dan evaluasi terhadap sistem ini juga dilakukan dengan menggunakan citra wajah yang berukuran 88 x 64 piksel, yaitu Basis Data wajah yang dimiliki oleh laboratorium riset YALE. Citra wajah Basis Data ini juga berada dalam

bentuk grey scale, dengan tingkat keabuan 256. Basis Data ini terdiri dari 165 obyek wajah dari 15 orang dengan 11 variasi untuk setiap orangnya. Data ini diindeks dengan angka 1 sampai dengan 11. Subyek bervariasi terhadap jenis kelamin, ekspresi wajah, pencahayaan (*lighting*) dan asesoris wajah (misalnya kacamata).

5.2.3 Hasil Ujicoba Data Whitening

Berikut ini akan diberikan hasil yang diperoleh dari percobaan yang dilakukan. Hasil dari pelatihan terhadap data pembelajaran dimasukkan kedalam data uji, kemudian hasilnya dihitung.

Tabel 5.8 Hasil pembelajaran Basis Data YALE dengan Metode ANFIS

Nomor Subyek	Leave-50%-out		Leave-75%-out		Leave-10%-out	
	Jumlah Data Pembelajaran	Jumlah Data Yang Dikenali Benar	Jumlah Data Pembelajaran	Jumlah Data Yang Dikenali Benar	Jumlah Data Pembelajaran	Jumlah Data Yang Dikenali Benar
1	5	5	8	8	9	9
2	5	5	8	8	9	9
3	5	5	8	8	9	9
4	5	5	8	8	9	9
5	5	5	8	8	9	9
6	5	5	8	8	9	9
7	5	5	8	8	9	9
8	5	5	8	8	9	9
9	5	5	8	8	9	9
10	5	5	8	8	9	9
11	5	5	8	8	9	9
12	5	5	8	8	9	9
13	5	5	8	8	9	9

14	5	5	8	8	9	9
15	5	5	8	8	9	9
JUMLAH		75		120		135
PROSENTASE		100%		100%		100%

Berikut ini akan diberikan hasil yang diperoleh dari percobaan yang dilakukan.

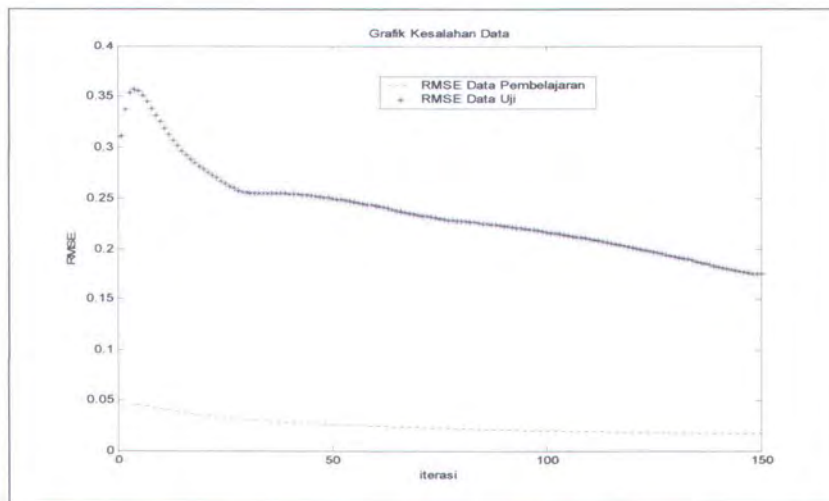
Parameter premis dan konsekuen yang dihasilkan di proses pelatihan terhadap

data pembelajaran dimasukkan kedalam data uji, kemudian hasilnya dihitung.

Tabel 5.9 Hasil pengujian Basis Data YALE dengan Metode ANFIS

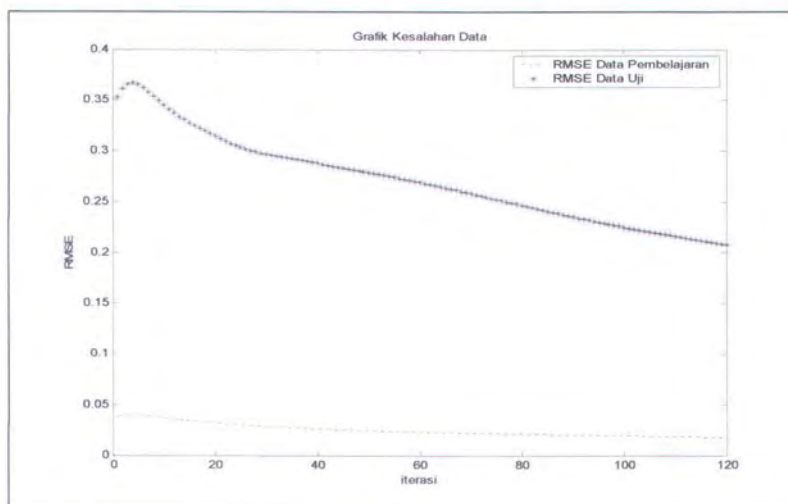
Nomor Subyek	Leave-50%-out		Leave-75%-out		Leave-10%-out	
	Jumlah Data Uji	Jumlah Data Yang Dikenali Benar	Jumlah Data Uji	Jumlah Data Yang Dikenali Benar	Jumlah Data Uji	Jumlah Data Yang Dikenali Benar
1	6	3	3	3	2	2
2	6	6	3	3	2	2
3	6	6	3	3	2	2
4	6	6	3	3	2	2
5	6	6	3	3	2	2
6	6	6	3	3	2	1
7	6	6	3	3	2	2
8	6	6	3	3	2	2
9	6	5	3	3	2	2
10	6	0	3	3	2	2
11	6	4	3	3	2	2
12	6	0	3	3	2	2
13	6	5	3	3	2	2
14	6	3	3	3	2	2
15	6	6	3	2	2	2
JUMLAH		68		44		30
PROSENTASE		75.55%		97.77%		96.66%

Dari pembelajaran dan pengujian tersebut didapatkan rata-rata kesalahan. Untuk data dengan metode pengambilan data *Leave-10%-out*, grafik RMSE dapat dilihat pada gambar 5.6

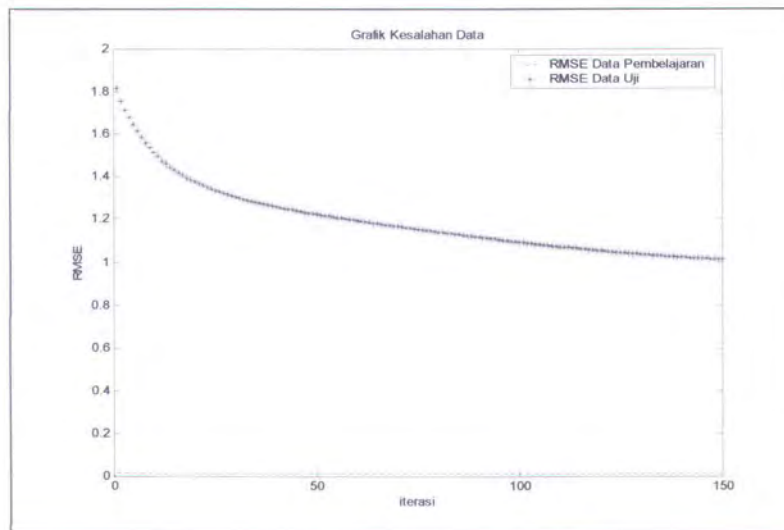


Gambar 5.6. Grafik RMSE untuk Data YALE Leave-10%-out

Sedangkan untuk data dengan metode pengambilan data *Leave-25%-out* dan *Leave-50%-out*, grafik RMSE dapat dilihat pada gambar 5.7 dan gambar 5.8



Gambar 5.7. Grafik RMSE untuk Data YALE Leave-25%-out



Gambar 5.8. Grafik RMSE untuk Data YALE Leave-50%-out

5.2.4 Hasil Ujicoba Data tanpa Whitening

Berikut ini akan diberikan hasil yang diperoleh dari percobaan untuk data yang tidak di-*whitening*. Pengujian dilakukan terhadap data pembelajaran dan data uji, kemudian hasilnya dihitung.

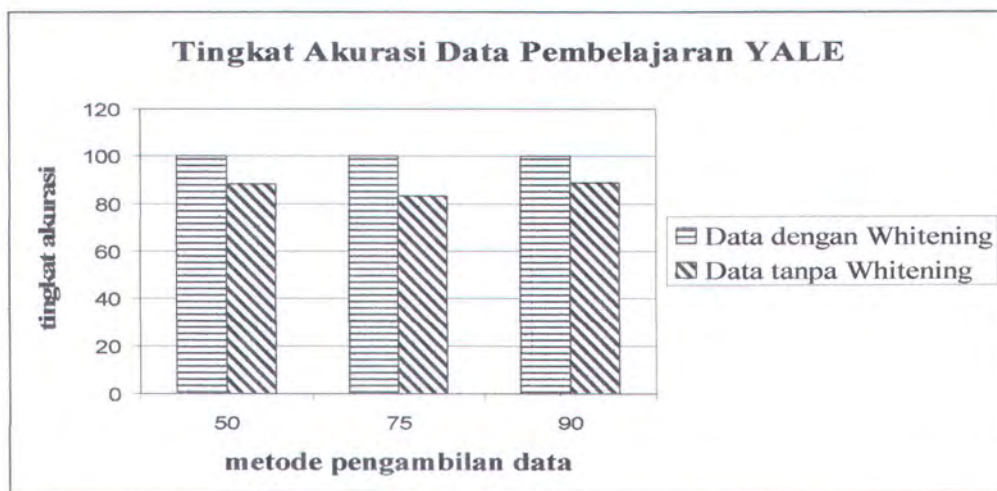
Tabel 5.10 Hasil pembelajaran Basis Data YALE (tanpa whitening) dengan Metode ANFIS

Nomor Subyek	Leave-50%-out		Leave-75%-out		Leave-10%-out	
	Jumlah Data Pembelajaran	Jumlah Data Yang Dikenali Benar	Jumlah Data Pembelajaran	Jumlah Data Yang Dikenali Benar	Jumlah Data Pembelajaran	Jumlah Data Yang Dikenali Benar
1	5	5	8	8	9	9
2	5	5	8	8	9	9
3	5	5	8	6	9	9
4	5	4	8	6	9	9
5	5	5	8	8	9	8
6	5	5	8	8	9	8

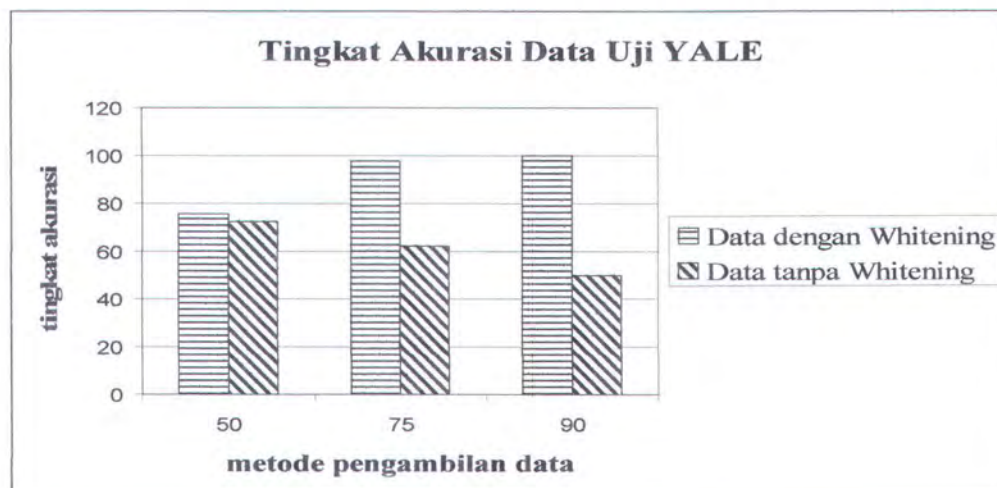


13	6	4	3	1	2	1
14	6	1	3	1	2	1
15	6	3	3	3	2	2
JUMLAH		65		28		15
PROSENTASE		72.22%		62.22%		50%

Perbandingan hasil ujicoba antara data yang di-*whitening* dengan data yang tidak di-*whitening* pada Basis Data YALE dapat dilihat pada gambar 5.9 dan 5.10 dibawah ini:



Gambar 5.9. Grafik Akurasi Data Pembelajaran YALE



Gambar 5.10. Grafik Akurasi Data Uji YALE

5.2.2.3 Uji coba dan Evaluasi terhadap Basis Data BERN

Selain dua Basis Data diatas, Basis Data yang digunakan untuk ujicoba dan evaluasi terhadap sistem ini adalah Basis Data BERN, yang dimiliki oleh The University of Bern (UB). Basis Data ini terdiri dari citra wajah yang berukuran 88 x 64 piksel dan berada dalam bentuk grey scale, dengan tingkat keabuan 256. Basis Data BERN memuat 28 orang yang masing-masing memiliki 10 citra wajah frontal. Basis Data BERN memiliki karakteristik adanya perubahan yang relatif kecil pada ekspresi wajah (*facial expression*) serta perubahan posisi kepala kearah kiri, kanan, atas dan bawah sebesar ± 30 derajat.

5.2.2 Hasil Ujicoba Data Whitening

Berikut ini akan diberikan hasil yang diperoleh dari percobaan yang dilakukan. Hasil dari pelatihan terhadap data pembelajaran dimasukkan kedalam data uji, kemudian hasilnya dihitung.

Tabel 5.12 Hasil pembelajaran Basis Data BERN dengan Metode ANFIS

Nomor Subyek	Leave-50%-out		Leave-75%-out		Leave-10%-out	
	Data Pembelajaran	Jumlah Data Yang Dikenali Benar	Data Pembelajaran	Jumlah Data Yang Dikenali Benar	Data Pembelajaran	Jumlah Data Yang Dikenali Benar
1	5	5	7	7	9	9
2	5	5	7	7	9	9
3	5	5	7	7	9	9
4	5	5	7	7	9	9
5	5	5	7	7	9	9
6	5	5	7	7	9	9
7	5	5	7	7	9	9

8	5	5	7	7	9	9
9	5	5	7	7	9	9
10	5	5	7	7	9	9
11	5	5	7	7	9	9
12	5	5	7	7	9	9
13	5	5	7	7	9	9
14	5	5	7	7	9	9
15	5	5	7	7	9	9
16	5	5	7	7	9	9
17	5	5	7	7	9	9
18	5	5	7	7	9	9
19	5	5	7	7	9	9
20	5	5	7	7	9	9
21	5	5	7	7	9	9
22	5	5	7	7	9	9
23	5	5	7	7	9	9
24	5	5	7	7	9	9
25	5	5	7	7	9	9
26	5	5	7	7	9	9
27	5	5	7	7	9	9
28	5	5	7	7	9	9
JUMLAH		140		196		252
PROSENTASE		100%		100%		100%

Berikut ini akan diberikan hasil yang diperoleh dari tiap percobaan yang dilakukan. Pengujian dilakukan terhadap data pembelajaran dan data uji, kemudian hasilnya dihitung.

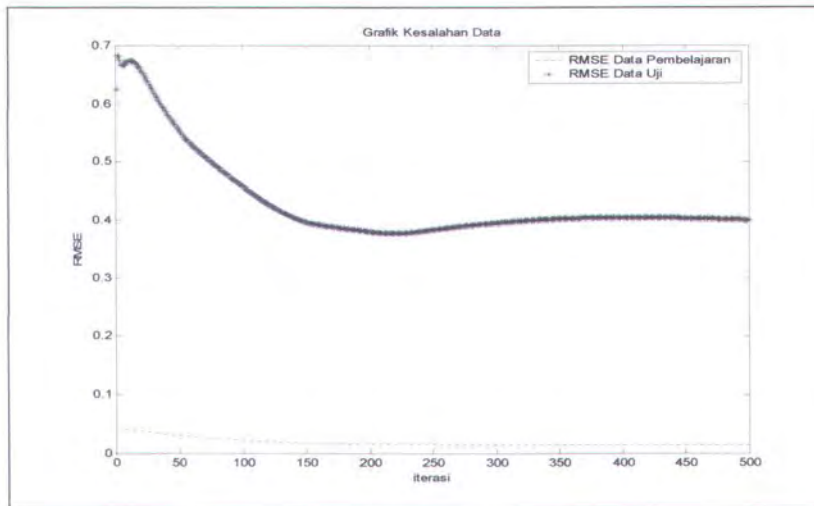
Tabel 5.13 Hasil pengujian Menggunakan Data BERN dengan Metode ANFIS

Nomor Subyek	Leave-50%-out		Leave-75%-out		Leave-10%-out	
	Data Uji	Jumlah Data Yang Dikenali Benar	Data Uji	Jumlah Data Yang Dikenali Benar	Data Uji	Jumlah Data Yang Dikenali Benar
1	5	2	3	1	1	1
2	5	5	3	3	1	1

3	5	5	3	3	1	1
4	5	3	3	2	1	1
5	5	1	3	2	1	1
6	5	3	3	2	1	1
7	5	3	3	1	1	1
8	5	4	3	3	1	1
9	5	3	3	2	1	1
10	5	3	3	3	1	1
11	5	5	3	3	1	1
12	5	5	3	3	1	1
13	5	0	3	2	1	1
14	5	5	3	3	1	1
15	5	5	3	3	1	1
16	5	3	3	3	1	1
17	5	5	3	3	1	1
18	5	5	3	3	1	1
19	5	5	3	3	1	1
20	5	5	3	3	1	1
21	5	5	3	3	1	1
22	5	5	3	3	1	1
23	5	5	3	3	1	1
24	5	5	3	3	1	1
25	5	5	3	3	1	1
26	5	3	3	2	1	1
27	5	5	3	3	1	1
28	5	5	3	3	1	1
JUMLAH PROSENTASE		113		74		28
		80.71%		88.09%		100%

Dari pembelajaran dan pengujian tersebut didapatkan rata-rata kesalahan (RMSE).

Untuk data dengan metode pengambilan data *Leave-10%-out*, grafik RMSE dapat dilihat pada gambar 5.11. Sedangkan untuk data dengan metode pengambilan data



Gambar 5.13. Grafik RMSE untuk Data BERN Leave-50%-out

5.2.3 Hasil Ujicoba Data tanpa whitening

Berikut ini akan diberikan hasil yang diperoleh dari percobaan untuk data yang tidak di-*whitening*. Pembelajaran dan pengujian terhadap data BERN tanpa *whitening* dapat dilihat pada tabel 5.14 dan 5.15

Tabel 5.14 Hasil pembelajaran Basis Data BERN (tanpa whitening) dengan Metode ANFIS

Nomor Subyek	Leave-50%-out		Leave-75%-out		Leave-10%-out	
	Data Pembelajaran	Jumlah Data Yang Dikenali Benar	Data Pembelajaran	Jumlah Data Yang Dikenali Benar	Data Pembelajaran	Jumlah Data Yang Dikenali Benar
1	5	5	7	7	9	9
2	5	5	7	7	9	9
3	5	5	7	5	9	3
4	5	5	7	7	9	3
5	5	3	7	2	9	8
6	5	3	7	2	9	7

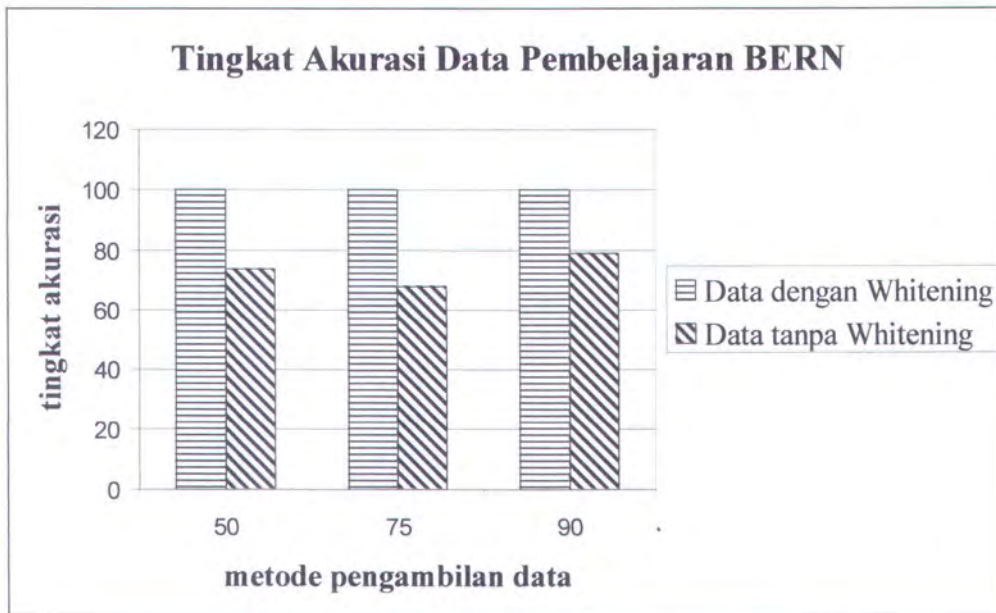
7	5	4	7	7	9	5
8	5	5	7	4	9	8
9	5	3	7	2	9	6
10	5	3	7	6	9	8
11	5	4	7	7	9	8
12	5	4	7	5	9	9
13	5	3	7	3	9	7
14	5	4	7	2	9	6
15	5	5	7	4	9	7
16	5	3	7	3	9	9
17	5	4	7	3	9	7
18	5	5	7	3	9	2
19	5	3	7	1	9	8
20	5	2	7	3	9	9
21	5	4	7	3	9	8
22	5	3	7	1	9	3
23	5	2	7	1	9	4
24	5	3	7	1	9	5
25	5	4	7	3	9	6
26	5	3	7	0	9	4
27	5	2	7	0	9	0
28	5	4	7	3	9	4
JUMLAH		103			133	199
PROSENTASE		73.57%			67.86 %	78.9 %

Tabel 5.15 Hasil pengujian Menggunakan Data BERN (tanpa whitening) dengan Metode ANFIS

Nomor Subyek	Leave-50%-out		Leave-75%-out		Leave-10%-out	
	Data Uji	Jumlah Data Yang Dikenali Benar	Data Uji	Jumlah Data Yang Dikenali Benar	Data Uji	Jumlah Data Yang Dikenali Benar
1	5	4	3	2	1	1
2	5	4	3	2	1	0
3	5	3	3	0	1	0

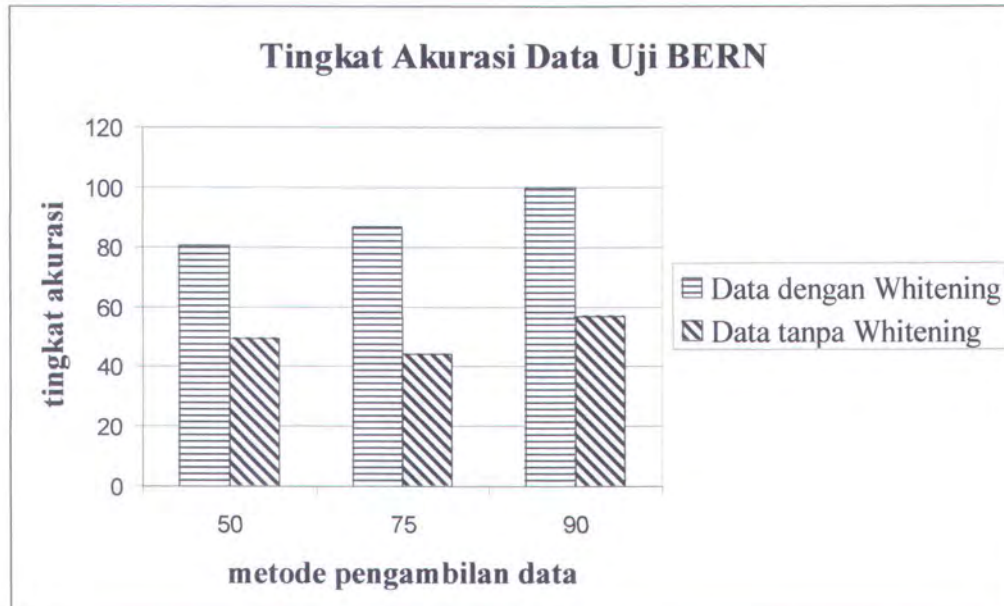
4	5	2	3	1	1	0
5	5	4	3	1	1	1
6	5	3	3	1	1	1
7	5	2	3	2	1	0
8	5	5	3	3	1	1
9	5	3	3	2	1	1
10	5	3	3	3	1	1
11	5	2	3	3	1	1
12	5	4	3	2	1	1
13	5	2	3	3	1	0
14	5	1	3	2	1	1
15	5	0	3	0	1	0
16	5	3	3	2	1	0
17	5	3	3	1	1	1
18	5	4	3	0	1	0
19	5	1	3	0	1	0
20	5	2	3	3	1	1
21	5	1	3	1	1	1
22	5	3	3	0	1	0
23	5	2	3	1	1	1
24	5	0	3	1	1	1
25	5	2	3	0	1	0
26	5	0	3	0	1	1
27	5	2	3	3	1	1
28	5	4	3	0	1	0
JUMLAH PROSENTASE		69		39		16
		49.28 %		46.42%		57.14%

Perbandingan hasil ujicoba antara data yang di-*whitening* dengan data yang tidak di-*whitening* pada Basis Data BERN dapat dilihat pada gambar 5.14 dan 5.15 dibawah ini.



Gambar 5.14. Grafik tingkat Akurasi Data Pembelajaran BERN

Untuk tingkat akurasi data ujinya dapat dilihat pada gambar dibawah ini:



Gambar 5.15. Grafik tingkat Akurasi Data Uji BERN

*Anas r.a berkata: Rasulullah SAW Bersabda: Siapa yang keluar untuk
menuntut ilmu maka ia berjuang fisabilillah hingga kembali – Attarmidzi*

BAB VI

KESIMPULAN DAN SARAN

BAB VI

KESIMPULAN DAN SARAN

Pada bab ini diberikan kesimpulan dan saran yang diambil berdasarkan pada ujicoba yang telah dilakukan.

1.1 KESIMPULAN

Dari aplikasi yang telah dibuat beserta uji coba yang telah dilakukan terhadapnya, penulis dapat mengambil kesimpulan sebagai berikut:

1. Metode ANFIS dapat diterapkan sebagai mesin pembelajaran dalam pengenalan pola, khususnya pengenalan wajah.
2. Kombinasi antara metode PCA dan Wavelet dapat diterapkan sebagai Fitur Ekstraktor untuk mengekstraksi fitur global dan lokal pada sistem pengenalan wajah.
3. Transformasi Whitening melibatkan basis non-orthogonal-nya sehingga dapat meningkatkan jangkauan ruang solusi. Dengan memperluas jangkauan solusi, tingkat kesalahan dapat direduksi sehingga dapat meningkatkan akurasi percobaan.



6.2 SARAN

Saran yang dapat diberikan penulis untuk pengembangan tugas akhir ini adalah:

1. Pengembangan aplikasi dalam hal file data yang dapat dibaca.

Dalam tugas akhir ini, file data yang dapat dibaca hanya dalam format text. Hal ini dapat dikembangkan dengan menambah fasilitas dalam aplikasi ini agar aplikasi tersebut dapat melakukan pembacaan data dalam format lainnya, misalnya dapat membaca file yang disimpan di Basis Data.

2. Penambahan metode pengambilan data

Pada aplikasi yang telah dibuat oleh penulis, pengambilan data hanya terbatas untuk yaitu *leave-10%-*, *leave-25%-out* dan *leave-50%-out*. Hal ini dapat dikembangkan dengan menggunakan metode pengambilan data lain, dengan prosentase pengambilan data yang berbeda, misalnya *leave-17%-out*, sesuai dengan keinginan pengguna. Atau dengan jumlah pengambilan data yang berbeda, misal *leave-80%-out*, yang berarti meninggalkan 80% data untuk testing data

3. Melakukan uji coba secara intensif dengan melakukan perubahan step size, untuk mengetahui angka berapa pada step size yang akan menghasilkan tingkat akurasi paling tinggi

DAFTAR PUSTAKA

- [1] Fang, Y., Wong Y. dan Tan T.: *Personalized Feature Combination for Face Recognition*.(2000).
- [2] Jang, J.-S R.: *ANFIS: Adaptive-Network based Fuzzy Inference System*, *IEEE trans. On System, Man, and Cybernetics*,
- [3] Soelaiman, Rully: *Sistem Pengenalan wajah dengan penerapan Algoritma Genetika pada Optimasi Basis Eigenface dan Proyeksi Fisherface*, Tesis: Program Pascasarjana Universitas Indonesia, 2003.
- [4] Damayanti, Fitri: *P3L Pengenalan Wajah menggunakan Metode Fisherface*, 2001.
- [5] Sarwosri: *P3L Klasifikasi Tekstur dengan menggunakan Analisa Paket Wavelet*, Tugas Akhir, 2000.
- [6] Ariani, Myke Noor: *Perbesaran Citra menggunakan Metode Wavelet*, Tugas Akhir, 2003.
- [7] Kusumadewi, Sri: *Analisis Desain Sistem Fuzzy menggunakan Tool Box Matlab*, Penerbit Graha Ilmu, Jogjakarta, 2002.
- [8] University of Bern, Bern, Switzerland, *University of Bern Face Database*, <ftp://iamftp.unibe.ch/pub/Images/FaceImages/>.
- [9] Yale Center for Computational Vision and Control, *Yale Face Database*, <http://cvc.yale.edu/projects/yalefaces/yalefaces.html>.
- [10] Garcia, C., Zikos, G. dan Tziritas, G.: *Wavelet Packet Analysis for Face Recognition*, Institute of Computer Science- Foundation for Research and Technology-Hellas (FORTH).

Dan jika kamu menghitung-hitung nikmat Allah, niscaya kamu tidak dapat menentukan jumlahnya. Sesungguhnya Allah benar-benar Maha Pengampun lagi Maha Penyayang. Dan Allah mengetahui apa yang kamu rahasiakan dan apa yang kamu lahirkan -- (An-Nahl 18,19)

Dan nikmat Tuhan kamu yang manakah yang kamu dustakan-(Ar-Rahman 13)



LAMPIRAN


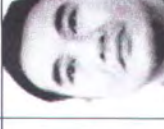


LAMPIRAN A









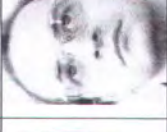
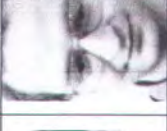
















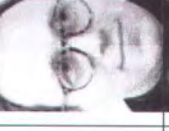
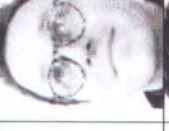


















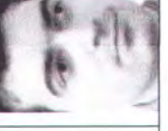


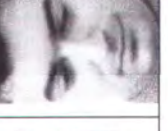
Basis Data Wajah University of Bern
















































Gambar yang digunakan untuk pelatihan maupun untuk uji coba pada Basis Data Wajah Bern [17]. Gambar tersebut terdiri dari 28 orang yang masing-masing memiliki 10 variasi pose.


























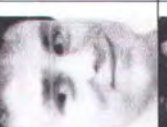

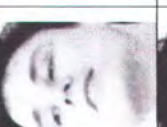
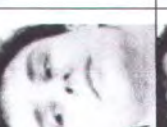






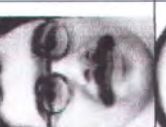









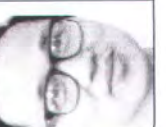
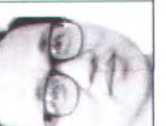





Pola	1	2	3	4	5	6	7	8	9	10
4										
5										
6										
7										
8										

Pola	1	2	3	4	5	6	7	8	9	10
9										
10										
11										
12										
13										

Pola	1	2	3	4	5	6	7	8	9	10
14										
15										
16										
17										
18										






















Pola	1	2	3	4	5	6	7	8	9	10
19										
20										
21										
22										
23										






















Pola	1	2	3	4	5	6	7	8	9	10
24										
25										
26										
27										
28										

LAMPIRAN B






















Basis Data Wajah Yale

Gambar yang digunakan untuk pelatihan maupun untuk uji coba pada Basis Data Wajah Yale [9]. Gambar tersebut terdiri dari 15 orang yang masing-masing memiliki 11 variasi pose.

Pola	1	2	3	4	5	6	7	8	9	10
1										
										
2										

Pola	1	2	3	4	5	6	7	8	9	10
2										
3										
4										

[illegible]

Pola	1	2	3	4	5	6	7	8	9	10
7										
8										
9										

[illegible]

[illegible]















































LAMPIRAN C



















































Basis Data Wajah OLLIVETTI



















































Gambar yang digunakan untuk pelatihan maupun untuk uji coba pada Basis Data Wajah OLLIVETTI. Gambar tersebut terdiri dari 40 orang yang masing-masing memiliki 10 variasi pose.



















































Pola	1	2	3	4	5	6	7	8	9	10
1										
2										
3										


































































































Pola	1	2	3	4	5	6	7	8	9	10
	4									
	5									
	6									
	7									
8										



















































Pola	1	2	3	4	5	6	7	8	9	10
9										
10										
11										
12										
13										










Pola	1	2	3	4	5	6	7	8	9	10
14										
15										
16										
17										
18										

Pola	1	2	3	4	5	6	7	8	9	10
19										
20										
21										
22										
23										

Pola	1	2	3	4	5	6	7	8	9	10
24										
25										
26										
27										
28										

Pola	1	2	3	4	5	6	7	8	9	10
	29									
	30									
	31									
	32									
	33									

Pola	1	2	3	4	5	6	7	8	9	10
34										
35										
36										
37										
38										

Pola	1	2	3	4	5	6	7	8	9	10
	39									
40	